Alain Sarlette January 6, 2009

Geometry and Symmetries in Coordination Control

PhD dissertation

Supervisor: Professor Rodolphe J. Sepulchre

Systems and Modeling Research Unit,
Department of Electrical Engineering and Computer Science,
Faculty of Applied Sciences,
University of Liège, Belgium.

Jury members:

Prof. L. Wehenkel (President)
Prof. R. Sepulchre (Supervisor)
Prof. P. Lecomte (University of Liège)
Prof. D. Aeyels (Ghent University)
Prof. V. Blondel (Université catholique de Louvain)
Prof. F. Bullo (University of California, Santa Barbara, USA)
Prof. P. Rouchon (Ecole Nationale Supérieure des Mines de Paris, France)

Abstract

The present dissertation studies specific issues related to the coordination of a set of "agents" evolving on a nonlinear manifold, more particularly a homogeneous manifold or a Lie group. The viewpoint is somewhere between control algorithm design and system analysis, as algorithms are derived from simple principles — often retrieving existing models — to highlight specific behaviors.

With a fair amount of approximation, the objective of the dissertation can be summarized by the following question: *Given a swarm of identical agents evolving on a nonlinear, nonconvex configuration space with high symmetry, how can you define specific collective behavior, and how can you design individual agent control laws to get a collective behavior, without introducing hierarchy nor external reference points that would break the symmetry of the configuration space?*

Maintaining the basic symmetries of the coordination problem lies at the heart of the contributions. The main focus is on the global geometric invariance of the configuration space. This contrasts with most existing work on coordination, where either the agents evolve on vector spaces — which, to some extent, can cover *local* behavior on manifolds — or coordination is coupled to external reference tracking such that the reference can serve as a beacon around which the geometry is distorted towards vector space-like properties. A second, more standard symmetry is to treat all agents identically.

Another basic ingredient of the coordination problem that has important implications in this dissertation is the reduced agent interconnectivity: each agent only gets information from a limited set of other agents, which can be varying.

In order to focus on issues related to geometry / symmetry and reduced interconnectivity, individual agent dynamics are drastically simplified to simple integrators. This is justified at a "planning" level. Making the step towards realistic dynamics is illustrated for the specific case of rigid body attitude synchronization.

The main contributions of this dissertation are

I. an extensive study of synchronization on the circle, (a) highlighting difficulties encountered for coordination and (b) proposing simple strategies to overcome these difficulties;

II. (a) a geometric definition and related control law for "consensus" configurations on compact homogeneous manifolds, of which synchronization — all agents at the same point — is a special case, and (b) control laws to (almost) globally reach synchronization and "balancing", its opposite, under general interconnectivity conditions;

III. several propositions for rigid body attitude synchronization under mechanical dynamics;

IV. a geometric framework for "coordinated motion" on Lie groups, (a) giving a geometric definition of coordinated motion and investigating its implications, and (b) providing systematic methods to design control laws for coordinated motion.

Examples treated for illustration of the theoretical concepts are the circle S^1 (sometimes the sphere S^n), the rotation group $SO(n)$, the rigid-body motion groups $SE(2)$ and $SE(3)$ and the Grassmann manifolds $Grass(p, n)$. The developments in this dissertation remain at a rather theoretical level; potential applications are briefly discussed.

Acknowledgments

Like for any piece of work distributed over a long period of time, achieving the present thesis would not have been possible without a caring, assisting and distracting life environment, which was masterly provided by family and friends; they are all most heartfully acknowledged for their support and patience in difficult times, as well as the enjoyable moments spent together.

Many thanks are also due to people of the working environment at Liège University. Fellow PhD students and other researchers of the department triggered interesting discussions during lunch time; one major result of this thesis is a hopefully lasting friendship with Michel Journée, Boris Defourny, Emre Tuna and Silvère Bonnabel. Inspiring discussions on subjects related to the present work involved many researchers in- and outside Liège University, among which Denis Efimov, Julien Hendrickx (UCL, Belgium), Pierre-Antoine Absil (UCL, Belgium), Maher Moakher (National Engineering School, Tunis), Thomas Krogstad (NTNU Trondheim, Norway) and others. Closer collaboration with postdoctoral researchers Luca Scardovi, Emre Tuna and Silvère Bonnabel was of great help, both scientifically and personally.

Another major experience was a 3-months visit at Princeton University in the research group of Professor Naomi Leonard. Related thanks at Liège University are due mainly to Professor Rodolphe Sepulchre for all contacts, arrangements and support, and also to Professor Jean-Pierre Swings for making this stay possible in connection with the Odissea prize.

Princeton University itself and its Mechanical and Aerospace Engineering Department are gratefully acknowledged for hosting an inexperienced Belgian PhD student. Special thanks go to Professor Naomi Leonard herself. Interesting discussions with other people in the MAE staff further enriched this stay, as well as students in Princeton University and at MAE department who contributed to make this stay fruitful and enjoyable in all aspects; this includes regular BEvERages after work with friends of the MAE PhD/Masters program starting September 2006, the MAE soccer team and the Princeton University Orchestra. Subjects closely related to the present work were discussed with Daniel Swain and now-Professor Derek Paley. Financial support was partly provided by the 1st Odissea prize initiated by the Belgian Senate.

Most hearty thanks go to Professor Rodolphe Sepulchre, who proved that he acquired more talents than needed to be a great thesis supervisor. Most of this work is due to his outstanding advice and support. Moreover, it is difficult to imagine how these three years would have been without his enthusiasm, patience and personal friendliness.

The members of the Jury are acknowledged for their interest and time devoted to this work.

Major financial support was provided by an FNRS fellowship and the Belgian Network DYSCO (Dynamical Systems, Control, and Optimization), funded by the Interuniversity Attraction Poles Programme, initiated by the Belgian State, Science Policy Office.

Contents

1 Introduction **11**
1.1 Why collective behavior of individually controlled, interacting entities is studied. 11
1.2 Why global geometry and symmetry is an important issue. 13
1.2.1 Synchronization and coordinated motion in one dimension 13
1.2.2 Synchronization and coordinated motion in higher dimension 16
1.2.3 Multi-agent tasks related to synchronization and coordinated motion . 19
1.2.4 Conclusion and more about the relevance of symmetries 19
1.3 What is new with respect to the existing literature 21
1.4 Summary and Outline of the presentation 22
1.5 Publications 25

2 Mathematical preliminaries **27**
2.1 General notation and definitions 27
2.2 Fundamentals of graph theory 29
2.2.1 Representing a network of interconnected agents by a graph 29
2.2.2 Weighted graphs, algebraic representation and Laplacian 29
2.2.3 Graph connectivity and time-varying graphs 30
2.2.4 Some particular graphs 32
2.3 Convergence results used in the proofs 33
2.4 Lie groups and homogeneous manifolds 37
2.4.1 Lie groups 38
2.4.2 Homogeneous manifolds 40
2.4.3 Tangent vectors, distances and gradients on manifolds 41
2.4.4 Some particular Lie groups and homogeneous manifolds 43

I Synchronization on the circle **47**

3 Synchronization: from vector spaces to the circle **51**
3.1 Consensus on vector spaces 51
3.2 Extension to the circle 53
3.2.1 Discrete-time algorithm 54
3.2.2 Continuous-time algorithm 55
3.3 Convergence properties 57
3.3.1 Local synchronization like for vector spaces 57
3.3.2 Convergence to local equilibria for fixed undirected $\mathbb{G}$ 59

3.3.3 Some graphs ensure (almost) global synchronization 61
3.4 Obstacles to global synchronization 62
3.4.1 Stable equilibria different from synchronization for fixed graphs 62
3.4.2 Limit sets different from equilibrium 65
3.4.3 State-dependent graphs 68

4 Global synchronization algorithms 73
4.1 Modified interaction profile for fixed undirected graphs 73
4.1.1 General idea 73
4.1.2 Algorithm and stability proof 74
4.2 Introducing randomness in link selection 77
4.2.1 Algorithm description 77
4.2.2 Convergence analysis 78
4.2.3 Simulation results and convergence rate 81
4.3 Dynamic controller with auxiliary variables 84
4.3.1 Algorithm description 84
4.3.2 Convergence analysis 85

II Mean and consensus on compact homogeneous manifolds 89

5 Defining consensus configurations on manifolds 97
5.1 The induced arithmetic mean 98
5.2 Consensus 101
5.3 Consensus as minimizing a cost function 104

6 Consensus algorithms on manifolds 109
6.1 Gradient consensus algorithms 109
6.2 Global consensus on manifolds 113
6.2.1 Dynamic controller with auxiliary variables 113
6.2.2 Modified interaction profile 120
6.2.3 Introducing randomness in link selection 122

7 Attitude synchronization of mechanical systems 125
7.1 Synchronization of mechanical systems 126
7.1.1 Mechanical systems on vector spaces 126
7.1.2 Mechanical systems on $SO(3)$ 128
7.2 Consensus tracking 130
7.3 Energy shaping 135
7.3.1 An existing attitude synchronization algorithm 136
7.3.2 Extension 1: relative angular velocities 137
7.3.3 Extension 2: directed and varying graphs 140

III Coordinated motion on Lie groups 145

8 Definitions of "coordinated motion" and their consequences 151
8.1 Symmetries, relative positions and coordination 151

8.2 Velocities and coordination . . . 152
8.2.1 Relative positions compatible with TC at a particular velocity . . . 154
8.2.2 Velocities compatible with TC and particular relative positions . . . 154
8.2.3 Changing velocities and relative positions in coordinated motion . . . 155
8.2.4 Using total coordination to define specific configurations . . . 157
8.2.5 Examples . . . 157

9 Designing control laws to stabilize coordinated motion 163
9.1 Coordination by consensus in the Lie algebra . . . 164
9.1.1 Right-invariant coordination . . . 164
9.1.2 Left-invariant coordination in the fully actuated setting . . . 164
9.1.3 Underactuated LIC and total coordination . . . 165
9.2 Control algorithms for fully actuated total coordination . . . 165
9.2.1 Total coordination on general Lie groups . . . 165
9.2.2 Total coordination on Lie groups with a bi-invariant metric . . . 167
9.3 Control algorithms for underactuated left-invariant coordination . . . 170
9.4 Coordinated motion with particular configurations . . . 174
9.4.1 Fully actuated agents . . . 174
9.4.2 Underactuated agents . . . 176

IV Conclusion 181
C.1 Summary . . . 183
C.2 Relevance in applications . . . 186
C.3 Leads for future work . . . 188
C.4 General lessons . . . 191

Appendix 195
A.1 Proof of Proposition 3.3.5 . . . 197
A.2 Lemma for the proof of Proposition 6.1.5 . . . 198
A.3 Bound for the proof of Proposition 7.2.2 (a) . . . 199
A.4 Details in the proof of Proposition 7.3.2 (b) . . . 199

Bibliography 203

Chapter 1

Introduction

The purpose of the present chapter is to give an overview of the dissertation without entering mathematical details. To this effect, the main ideas are progressively introduced by intuitive examples. This is also an opportunity to loosely define some vocabulary for fundamental notions in this dissertation. A first section introduces the notions of swarm and collective behavior, and briefly discusses the relevance of a control setting where individual agents decide on their actions based on locally available information, with no supervisor or common reference explicitly coordinating the swarm. A second section illustrates the focus of the present dissertation: it presents the tasks addressed by "coordination" and the related central importance of geometry and symmetry; it also draws a link between the presented coordination tasks and related questions involving multiple agents. A third section broadly reviews problems and solutions known in the literature; more detailed literature reviews per topic can be found in the main body of the dissertation. A fourth section summarizes the main contributions in the present document and explains its further organization. The chapter concludes with a list of publications presenting the results of this work.

1.1 Why collective behavior of individually controlled, interacting entities is studied.

A *swarm* of individually controlled and interacting entities, or *agents*, has the following properties.

- *Individual ("decentralized") control:* each entity controls its motion individually, in order to achieve its individual objective in the best possible way, without being told what to do by any potential "supervising coordinator".
- *Limited interaction:* each entity can communicate with a limited number of other entities. This allows the entity to take the behavior of its fellows into account when formulating its individual objective.
- *Collective behavior:* if individual behaviors (defined mainly by the objectives) are chosen appropriately, the swarm can behave like a coordinated unit which can achieve a common objective.

Swarms can be advantageous for various reasons, see Figure 1.1.

- Distributed search for information: The advantage of exploring a large domain with a swarm of organized individuals is exploited for animal foraging (see [33, 43]), robotic

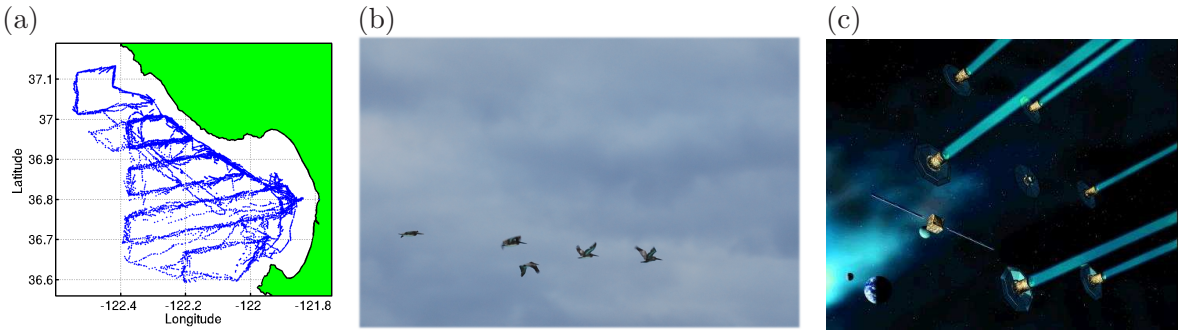


Figure 1.1: Applications involving swarms of individuals: (a) Trajectories of underwater gliders gathering ocean data in a distributed way (project by Prof. N. Leonard, Princeton University). (b) Birds flying in formation over Tulum, Mexico. (c) Artist's view of the Darwin space interferometer project (European Space Agency).

exploration of unknown planets (see [39]), or distributed database maintenance e.g. in ocean exploration or surveillance (see [29, 76]).

- *L'union fait la force* ("Unity is strength"): Individual agents can gain benefit from travelling together in a coordinated formation. Examples include birds and fish travelling together for energy efficiency — e.g. like in team time trials of the Tour de France cycling race — and for better defense against predators (see [31] and references therein), ants or robots working together to carry a large/heavy load, or the formation flights and platoon movements introduced mainly for military operations (see [40, 142]).
- Assembly: Sometimes it is desired to have one big entity, but for practical reasons it cannot be provided as such; then a solution is to use several separated parts and coordinate them such that they can assemble (physically or in their function) for operation. Classical examples of this type are spacecraft assembly (see [55, 86]) and resolution enhancement with multiple coordinated telescopes through interferometry (see [9, 93]).

As can be seen from these examples, many recent engineering applications of swarms are inspired by the behavior of natural systems. Because the coordination of natural systems emerges from individual decentralized control, related phenomena are also called "emergent ordering" or "spontaneous ordering", or "order-disorder phase transitions" in physics. The study of emerging collective phenomena is a fascinating subject in itself, that has gained attention in the last decades in both the theoretically oriented communities of dynamical systems and statistical physics, as well as more practically oriented experimental physics and biology communities (see [32],[71],[138],[148] and references above).

In the setting of the present dissertation, the use of swarm control for engineering applications takes over the architecture of natural swarms: individual agents decide on their actions based on locally available information, with no supervisor or common reference explicitly coordinating them. An additional assumption, which seems plausible for natural systems as well, is that all agents behave equivalently, i.e. the form and parameter values in the control laws do not differ from one agent to another. This could be called an "*autonomous individuals*" - type swarm approach. Other approaches in the engineering literature consider

swarm control with a central supervisor which explicitly computes all the control laws to coordinate the agents (see [4] and many others), or with a common reference that all the agents track (see [111] and many others, often without explicitly saying so), or with different roles associated to different agents e.g. introducing a leader which decides on (or is assigned) a particular swarm behavior which the others — the followers — follow (see [15] and many others). Research around e.g. the Kuramoto model (see [71, 137]) investigates the robustness of collective phenomena with respect to differences among similar agents, studying how the interactions can coordinate different natural behaviors.

The choice of an "autonomous individuals" - type setting in the present dissertation is motivated by the following.

- The "autonomous individuals" architecture is more robust to failures than a supervised or leader-follower architecture. In the latter, if the supervisor or leader experiences a failure, then the whole swarm is doomed to fail or even collapse; in particular, reliability of the supervisor's communication capabilities is critical. In contrast, in an "autonomous individuals" swarm no individual is critical: when one agent drops out, there is just one less fellow to take into account, so the swarm can keep working and generally achieves only marginally decreased performance.
- The supervised approach has bad *scalability*: for an increasing number of agents in the swarm, the job of the supervisor becomes increasingly difficult, regarding both computation (the control decision task) and communication with all the members of the swarm. This problem does not appear in an "autonomous individuals" approach where each agent makes its own computations and communicates only with a restricted set of other agents.
- From a viewpoint of analysis rather than design, an "autonomous individuals" description is closer to natural collective phenomena of interacting entities in biology and physics; the hope is that examining issues in swarm control in the "autonomous individuals" setting may give additional insight to understand some of these phenomena.

Finally, it must be stressed that from a perspective of control law design and analysis, the "autonomous individuals" setting is an additional difficulty with respect to supervised or reference-following approaches, rather than a limitation: in presence of a common reference or supervisor, most of the core problems addressed in the following become rather trivial and coordination can be solved with existing results about nonlinear tracking control, like [22, 139].

1.2 Why global geometry and symmetry is an important issue.

1.2.1 Synchronization and coordinated motion in one dimension

A. Synchronization: Imagine N *agents* (persons, robots, particles,...) which must agree on a one-dimensional quantity of interest; in the words of the present document, they must *synchronize*. Starting each with its initial bet (opinion, information, state,...), the agents would move towards each other in the hope of finally — maybe *asymptotically*, after infinite time — reaching the same value.

Ex. 1.2.1: synchronization on the real line: Consider situations where N persons must agree on a desired room temperature, on the price at which a product will be sold or bought, or on the velocity at which to fly their spacecraft. These examples have the common feature

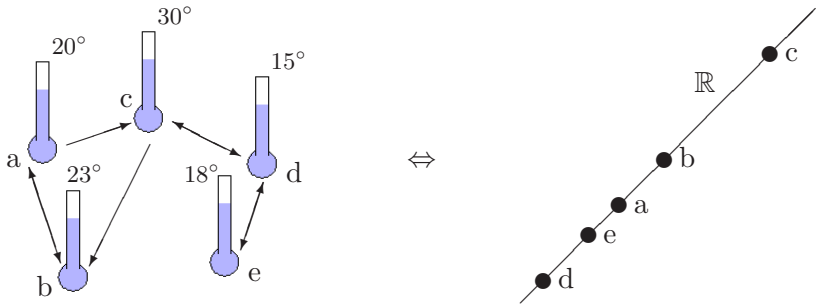


Figure 1.2: Synchronization on the real line: a group of interacting persons must reach agreement on a desired room temperature; arrows symbolize communication flow.

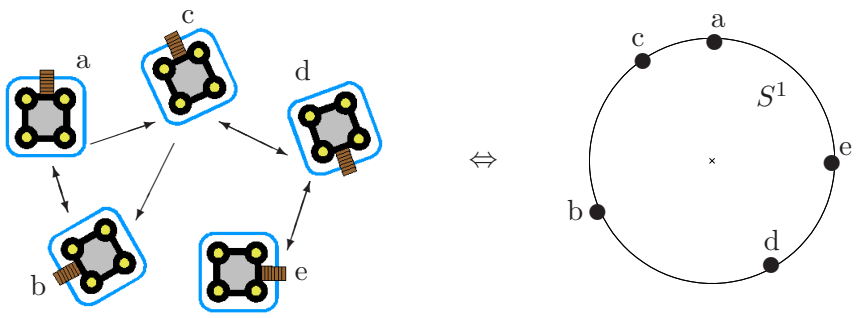


Figure 1.3: Synchronization on the circle: interacting persons must reach agreement on a desired house orientation (i.e. direction of entrance bridge); arrows symbolize communication.

that the quantity of interest takes values on the real line. It is analogous to getting cars on a straight track to meet at some point. See Figure 1.2. ◇

Ex. 1.2.2: synchronization on the circle: Imagine that N persons must reach a common decision on a direction of motion in the plane, on the orientation of a house (i.e. in which direction to point the entrance door), or on the time at which a daily meeting will take place. In these examples, the quantity of interest is cyclic: initially orienting the house towards the North and starting to turn it clockwise to check out alternatives, one will see it pointing eastwards, southwards, westwards and finally northwards again. Synchronization of cyclic variables is like getting cars to meet at some point on a closed (without loss of generality, circular) racetrack. See Figure 1.3. ◇

The space on which the variables evolve is called the *configuration space*. The examples illustrate the importance of the *global geometry* — actually *topology* — of the configuration space. For synchronization on the real line, the agent with the highest (respectively lowest) value will always decrease (resp. increase) its value in order to move towards its fellows, such that one reasonably expects them to meet at some point in the middle of their initial bets. For synchronization on the circle, the situation is similar when agents are close together. However, when agents are distributed over the whole circle, it is impossible to distinguish

a "highest" and "lowest" value, and a point "in the middle of" the initial bets; in terms of the examples, a temperature of 20° is certainly between 18° and 35°, but it is impossible to decide if the point "between" East and West is North or rather South. This can be formulated differently by considering the agents' motions: on the circle, an agent that gets further apart from a group of fellows by constantly moving in one direction can eventually meet them again after one turn — faster cars on a closed racetrack keep overtaking the slower ones — while on the line it would just escape once and for all. As a conclusion, synchronization behavior is a priori not as clear on the circle as on the real line.

Symmetry, or *invariance*, is important in synchronization problems.

- Invariance with respect to individual agents: In a democratic world[1], there is no a priori reason for one agent to have more authority than others: the outcoming agreement should be the same if A wants room temperature / house orientation α and B wants β, or if A wants β and B wants α. A variant associates weights to the agents to reflect their relative importance, but in any case agents of equal weight should be treated identically.
- Invariance with respect to "absolute" values: In many cases, there is no a priori preferred value for the quantity of interest; then the behavior of the group should be exactly similar if all agents translate their value by a constant quantity. Vehicles in outer space have no reference absolute velocity, so if each commander wants to fly say $1km/s$ faster, then the agreement velocity will be $1km/s$ faster. Similarly, to agree on a daily teleconference time for people scattered around the globe, there is no reason to prefer one particular time over another; so if each individual shifts its desired meeting time by one hour, the agreed time will be shifted by one hour. This is like synchronizing the positions of cars on a closed circular racetrack without pitlane, or on a roundabout without any exit: there is no way to define a preferred *absolute* position on the circular track, preferences of a car can only be with respect to its current position.

B. Coordinated motion: Instead of aspiring to a common agreement value, N agents may just want to move on the configuration space in some "coordinated" way. In one-dimensional space, there is not much choice to define coordinated motion: the *velocity of the agents on the configuration space* must be the same, such that their relative positions remain constant. The agents are said to *move in formation* and a particular set of relative positions of the agents is called a *formation* or *configuration*.

Ex. 1.2.3: coordinated motion on the line: Revisiting the applications of Example 1.2.1, one may want to keep constant price differences between products, or constant temperature differences between compartments, regardless of the evolution of absolute price or temperature. It is not clear in which situations one could like to maintain constant velocity differences among spacecraft; however in this case, as the configuration space is *linear velocity*, the agents will have to agree on a common *linear acceleration* (which is the velocity on configuration space in this case). A more appealing application of coordinated motion would have *linear position* (say, altitude or position on a linear race-track) as configuration space — for instance, several agents carrying a single solid object should "move in formation" in the sense of maintaining constant linear position differences. They therefore must agree on the

[1]The physical laws are democratic, although not exactly in the political sense: particles are treated equivalently if they are *exactly identical*. The difference with politics is that just being electrons does not guarantee equal votes, as the latter also depend on the energy they own, their spin,... .

same *linear velocity*. ◇

Ex. 1.2.4: coordinated motion on the circle: Keeping constant orientation/direction differences amounts to agreeing on a rotation rate. This can be used for instance to impose equal speeds on the four (or more) wheels of a vehicle. Coordinated motion of timing devices is critical, as it means that they "measure time as elapsing at same speed", i.e. the indexes of the clocks move at the same speed; the *Kuramoto model* [71, 137] was a first benchmark to study clock coordination. Cars on a circular race-track may want to maintain distances fixed in order to avoid collision or, as on the linear track, to carry a single solid object. ◇

Symmetry takes even more importance in the framework of coordinated motion: the group must behave independently of the current absolute position on the configuration space, while the agents are actually *moving*, thus visiting different absolute positions.

Regarding geometry, on the line, synchronization and coordinated motion are basically the same problem. This is best illustrated by the fact that, from an abstract point of view, synchronization of linear velocities in Example 1.2.1 is in fact the same problem as coordinated motion of linear positions in Example 1.2.3. There may be small differences about the way the actual agents are controlled, but they are disregarded here. In contrast, coordinated motion is not similar to synchronization on the circle. In fact, coordinated motion on the circle rather looks like synchronization on the line: if agents want to move as a formation, they must agree on a rotation velocity around the circle, or instantaneous frequency, which takes values on the real line. However, on higher dimensional configuration spaces, the situation can become more complex and coordinated motion does not simply reduce to synchronization on a vector space.

1.2.2 Synchronization and coordinated motion in higher dimension

Ex. 1.2.5: synchronization and coordinated motion on vector spaces: Consider for instance a set of spacecraft evolving in 3-dimensional space, a set of fish moving in the 3-dimensional ocean, or a set of ship moving on the surface of a lake. They may want to gather at some point, which requires to agree on a desired meeting position in 3-dimensional space or in the plane (2-dimensional space). Alternatively, they may want to move in coordinated motion as a formation, i.e. keeping fixed distances between agents; this can be useful to systematically explore an area, to cooperatively carry a common solid load, or as a simple means of avoiding collisions while travelling. Such coordinated motion requires to agree on a value in 3-dimensional space or in the plane to be the common desired velocity of all the agents. ◇

The plane and 3-dimensional space are examples of *Euclidean spaces* or *vector spaces:* they are "flat" and differences among positions in such spaces are canonically defined by vector differences. The previous examples illustrate that, unsurprisingly, both agreement problems of synchronization and coordinated motion are equivalent on Euclidean spaces. This is in fact a direct consequence of the fact that they are "flat", such that the tangent space — on which the velocity is defined — at any point of the configuration space is actually equivalent to the configuration space itself. Conceptually, the coordination problems on vector spaces are strictly equivalent to the case of the line: the procedure for the line is simply applied to every coordinate direction.

Ex. 1.2.6: synchronization on nonlinear manifolds: Consider a decision on the position of a radar station on Earth; this requires to reach agreement on a point of the sphere. A similar task is required for the meeting of a set of spacecraft on orbit at a fixed altitude around some planet, or of vehicles, animals or humans distributed over the whole surface of the Earth. As another example, consider a set of fish, submarines or spacecraft which can move in three-dimensional space but always with a fixed speed, with only the possibility to choose their direction of motion. If they want to achieve coordinated motion along a straight line, they have to agree on a direction of motion in 3 dimensions, which is equivalent to agreeing on a point of the sphere. The sphere has dimension 2. Going one step further, consider a set of agents which want to have the same orientation in 3-dimensional space; this could be telescope satellites which want to get exactly the same view of some object, molecules which must have particular orientations in order to assemble,... . These agents have to agree not only on a heading in 3 dimensions (i.e. a point on the sphere), but also on their orientation around this heading vector: they must achieve synchronization on the 3-dimensional set of all possible 3-dimensional orientations, equivalent to the set of all 3-dimensional rotation matrices. ⋄

Reaching agreement on a point presents similar problems on the sphere as on the circle: when the agents are distributed over the whole configuration space, it is hard to say how the system will behave when the agents move "towards their fellows", since every agent sees other agents in all directions around itself. A similar problem occurs for the 3-dimensional rotation matrices. These spaces, on which synchronization has a problematic behavior that does not appear on vector spaces, are "curved"; they are called *nonlinear manifolds*, or often just *manifolds* — although vector spaces are also manifolds, but trivial linear ones.

Synchronization on manifolds can in fact be necessary to achieve coordinated motion on vector spaces when the agents are underactuated, i.e. when their motion in the vector space is restricted. However, to be rigorous, for the restriction to fixed speed to make sense in the applications cited in Example 1.2.6, the problem setting should be enlarged to include not only positions but also orientations of the agents in the configuration space. This leads to a problem of coordinated motion on *Lie groups*, which are a particular class of nonlinear manifolds. The following example uses the sphere to illustrate the problem of coordinated motion on nonlinear manifolds. The sphere is not a Lie group. In fact the intuitive discussions implicitly involve an orientation variable, such that the actual configuration space is not the sphere, but the Lie group $SO(3)$; however the visualization is greatly simplified by not explicitly mentioning the orientation variable. The illustrated definitions are exactly those of Part III, where Example 8.2.10 clarifies the case of the sphere.

Ex. 1.2.7: coordinated motion on the sphere: Consider a set of agents moving on the sphere. Now try to define what it means for those agents to "have the same velocity". A first attempt could be to consider the velocity vectors in 3-dimensional space. The possible velocity vectors of an agent belong to the plane that is tangent to the sphere at the position of that agent. But taking agents at different positions, the tangent planes are all oriented differently, and in general for more than two agents there is no common nonzero vector belonging to their intersection, see Figure 1.4. Thus defining coordinated motion with equal velocity vectors is not possible. In fact, this is also the case on the circle: agents located at different positions on the circle do not have the same velocity vector during coordinated motion as defined in Section 1.2.1.

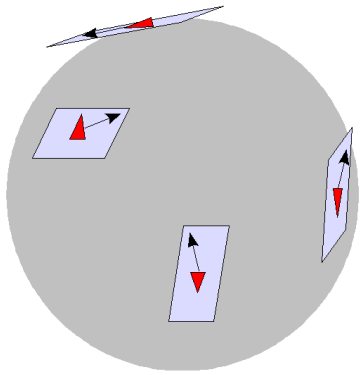


Figure 1.4: Agents (triangles on the picture) located at different positions on the sphere have linear velocities (arrows on the picture) belonging to different tangent planes. The intersection of the 2-dimensional subspaces of $\mathbb{R}^3$ to which the velocity of the different agents can belong, generally reduces to the origin, i.e. zero velocity.

One way to actually define coordinated motion is to require that the agents all draw the same curve on the circle, except that the curves may start at different initial points and in different directions; using the terminology as for "similar triangles", one would say that the agents draw *similar curves*. In other words, at all times, all the agents have "the same velocity in their local frames", in order to draw rotated/translated versions of the same trajectory.

Another way to define coordinated motion is to require that the relative positions of the agents stay fixed. In this case, velocities of the agents may even have different magnitudes during coordinated motion: if for instance several agents move at equal steady speed along the equator, another agent located at a different latitude will have to move at slower speed on a parallel (i.e. a circle of smaller diameter in a plane parallel to the equator), and an agent located at the pole will not be allowed to move at all.

Finally, one could want to combine both types of previously defined coordinated motions, requiring that both conditions "similar curves" and "fixed relative positions" are simultaneously satisfied. This kind of coordinated motion indirectly restricts possible relative positions. Indeed, consider for instance that, as previously, several agents move at steady speed along the equator. Then, it is not possible to add an agent elsewhere than on the equator, because a motion on a parallel of different diameter would not draw a trajectory similar to the equator; reasoning the other way, a trajectory similar to the equator would have to be a great circle, but if an agent moves on a circle which is tilted with respect to the equator, then its relative position with respect to the agents on the equator varies. ◇

The problem encountered in the first paragraph of Example 1.2.7 shows that coordinated motion on higher-dimensional manifolds is not simply equivalent to synchronization of velocities belonging to a vector space. Some way must be defined to compare velocities belonging to *different* tangent spaces. Example 1.2.7 proposes two solutions to define coordinated motion on the sphere, which illustrates that comparison of velocities can be done in different ways[2].

The situation of the last paragraph in Example 1.2.7, combining both types of coordinated

[2]On the circle, both solutions reduce to the same requirement: all agents must have the same rotation rate.

motion, is not feasible with arbitrary positions of the agents. This implies additional difficulty since, in contrast to coordinated motion on vector spaces, appropriate relative positions must be reached in addition to appropriate velocities for coordinated motion to be possible.

1.2.3 Multi-agent tasks related to synchronization and coordinated motion

In addition to reaching the same position (synchronization) and moving in a coordinated way, there are other problems related to the organization of a set of agents on a manifold.

A first related problem is the computation of a "mean" or average position of several points on a manifold. For instance, consider a set of points arbitrarily distributed on the circle, and try to pick a point on the circle which is the "mean position" of these points. This is not as obvious as computing the arithmetic mean of points on a line, plane or higher dimensional vector space. For some cases, it is even evident that no point can be singled out as the mean position — consider for instance uniformly distributed points, like 12 points placed exactly at the 12 hour-marks of a clock. The problem of defining a mean is directly related to synchronization, since the mean could serve as a meeting point.

Beyond synchronization, there could be other specific configurations of interest, like spreading the agents in some way for instance — the example of 12 points located on the circle at the 12 hour-marks is a particular configuration of this type. The associated tasks are to define such configurations, and to design appropriate control laws to reach them.

Regarding coordinated motion, applications often require not only to agree on some common way of motion — as in Example 1.2.7, tracing similar trajectories or moving with fixed relative positions — but also to build a specific "formation", where agents are located at particular relative positions with respect to each other.

Ex. 1.2.8: applications involving "formations": In practical applications of coordinated motion of real objects, relative positions cannot be completely arbitrary because at least collisions must be avoided. In some cases, keeping more specific relative positions is important. For instance, autonomous underwater vehicles moving as a swarm to collect ocean data must be distributed in such a way that local gradients can be accurately estimated. Specific "formations" are also often used for other strategic reasons, like the "triangle" formation often observed for groups of large birds or airplanes. Sometimes vehicles must even cope with physical links, like when cooperatively carrying a large solid object or for online refueling operations. ◇

1.2.4 Conclusion and more about the relevance of symmetries

The above examples highlight the presence and specificity of *nonlinear geometry* — sometimes arising from constraints — and related *symmetries* in coordination problems. The present dissertation studies coordination on nonlinear manifolds, like the circle as the simplest example, both in the sense of position agreement (synchronization and other specific configurations) and in the sense of coordinated motion ("velocity agreement" to be defined).

The focus is on solving *agreement problems*, where a set of agents decide on their behavior through interactions only, without the influence of an external reference. In such a setting, it is natural to consider that the agents' behavior should remain unchanged if they are "all translated in the same way". This is formalized by considering configuration spaces with "highly symmetric" geometry and agents with simple symmetric dynamics. In absence of an

external reference, there is no possibility for the agents to "break" the symmetry with respect to uniform translations on the configuration space: cars on a roundabout with no exit have no way to behave differently with respect to each other when they are all rotated in the same direction by, say a quarter-circle with respect to their initial position. This means that the corresponding symmetry must be maintained for all defined objects and control laws; this is a main requirement of the present dissertation.

Many authors, especially regarding the attitude synchronization problem discussed in Chapter 7, study coordination in conjunction with reference tracking. In this context, some more discussion about the relevance of an invariant setting, with no reference tracking, may be needed.

In engineering applications, coordinated swarms indeed often interact with an external reference — a path to follow, a specific beacon in the environment for autonomous vehicles, or a star or Earth station to point at for satellite formations,... . This is probably the main reason for the extensive literature about synchronization in presence of a common reference. However, one could argue that "coordinating" agents to track a common reference is only a marginal added value with respect to simply reference tracking: in the absence of perturbations, the agents do not even need to interact in order to reach coordination, since the latter is a direct consequence of tracking the same reference. In contrast, in an invariant setting with no external reference, the agents *have to* interact in order to reach agreement and achieve a collective behavior. Now it may be argued if such a more inherent coordination framework is useful at all. Arguments in favor of a positive answer to this question are given in Section 1.1 to justify the "autonomous setting". From the viewpoint of geometrical symmetries, the following can be added.

- An advantage of building controllers which achieve coordination invariantly with respect to absolute position is that tracking can be easily and independently added to the invariant coordination framework if necessary.
- In some settings, there may indeed be situations where the main problem is to reach agreement on some quantity. When for instance a "mean position" must be computed from a dataset on a manifold, a meaningful result should not depend on some arbitrary choice of reference coordinates.
- The invariant setting is better adapted to the understanding of basic coordination mechanisms. In the 3-dimensional physical world, the laws governing interactions in a set of particles are invariant with respect to (static) translations and rotations of the whole set as a rigid body. From this viewpoint, the symmetry assumption comes down to assuming that the agents are isolated (i.e. there is no external influence acting on the agents); in many natural phenomena, this assumption can be considered as satisfied.

Further discussion about the relevance of symmetry in realistic settings can be found in the Conclusion (Part IV, Section C.2).

Ex. 1.2.9: some highly symmetric spaces: The following highly symmetric spaces are more or less explicitly considered in the present dissertation: Lie groups, including the group $SO(n)$ of n-dimensional rotations — which specializes to "attitudes" (i.e. orientations in 3 dimensions, $SO(3)$) and "headings" (i.e. orientations in the plane, $SO(2)$ = the circle) — as well as the groups $SE(3)$ and $SE(2)$ of rigid body motion (i.e. translations and rotations) in 3 and 2 dimensions respectively; the Grassmann manifolds $Grass(p, n)$, on which a point

represents a p-dimensional subspace of $\mathbb{R}^n$; the sphere S^n, which is the set of points of $\mathbb{R}^{n+1}$ whose position vector has unit norm. $\diamond$

1.3 What is new with respect to the existing literature

Before starting to compare the present dissertation to the existing literature about coordination, apologies are due to the many authors which are active in this field but whose work is not cited in the present document. Unfortunately (or maybe fortunately), the interest in the field of coordination control has recently grown so large that a fair review of all related work is beyond the reach of a single document. The choice of citations in the present work is not meant to establish a hierarchy of important and less important results; it is mostly a consequence of the cited work's relation with the present approach, and sometimes (especially when illustrating the general background) it may appear to be arbitrary.

A first line of thought in the framework of coordination is the basic problem of reaching agreement among several agents, also known as the "consensus problem" or agreement problem. The agreement problem for variables belonging to a vector space has been extensively studied, with important convergence results including [13, 94, 95, 104, 105, 143]; see [102] for a review. However, few results are available for variables belonging to nonlinear manifolds. "Synchronization" of oscillators, involving phase variables on a configuration space isomorphic to the circle, is actually mostly concerned with reaching a common rotation rate — not a common phase — which as discussed in Section 1.2.1 actually involves agreement on the real line. The problem becomes significant in the presence of different natural dynamics for the individual agents, as in the Kuramoto model [71, 137]; studying the robustness of coordination with respect to divergent natural tendencies is another important issue, but beyond the scope of the present work. Some authors, like [57, 95], propose local results for the agreement problem on nonlinear manifolds; they are essentially identical to their vector space counterparts, because by definition a manifold can be locally mapped to a vector space. A few results about the synchronization of satellite orientations, like [96, 109, 135], can be said to tackle an agreement problem on 3-dimensional orientations — most other work actually relies on a common reference; although [96, 109, 135] take a fully nonlinear viewpoint, with mechanical dynamics, significant convergence results are still local. For the rest, there seems to be no attempts to study agreement problems on nonlinear manifolds.

The present work wants to propose a framework, algorithms and convergence results for agreement problems on nonlinear manifolds with arbitrary initial conditions. The last fact implies that the *global geometry* of the manifold must be taken into account. This main difference with respect to previous work gives raise to issues that are completely new with respect to agreement problems on vector spaces.

The related problem of mean computation on manifolds is well understood thanks to extensive studies in the applied mathematics community, see e.g. [6, 23, 41, 44, 52, 65, 68, 92, 108]. The present dissertation just examines a computationally simple alternative with respect to the canonical Karcher or Frechet mean on which this line of work mostly centers.

A large number of authors have recently proposed control laws for specific applications or settings involving coordinated motion — see as a small sample [4, 35, 53, 62, 63, 84, 89, 103, 123, 131, 141, 142] and references therein. Besides the most popular topic of stabilizing

specific formations, the objective of mobile sensor network control [29, 76] actually started a collaboration from which the present dissertation topic has derived. Most papers cited above concentrate on particular control laws associated to their specific goals and setting. Their ideas are based on physical properties like passivity, intuition and experience with Lyapunov-based design, or complex controllers combining several levels or effects. One popular subject in the last direction is for instance a behavioral approach implementing the "three flocking rules" of Reynolds [113]. Another line of work considers the coordination of *linear* systems, where classical control design tools are applied. However, even for applications involving motion of rigid bodies in a vector space, the body orientations (e.g. headings in 2 dimensions, attitudes in 3 dimensions) evolve on nonlinear manifolds.

The present dissertation proposes a *general geometric viewpoint* for coordinated motion on Lie groups, and allows arbitrarily distributed agents as a consequence of considering the global geometry. It thereby somewhat moves away (e.g. considering simplified dynamics) from the settings involved in realistic control applications, but allows to identify basic issues related to coordinated motion, drawing links between multiple specific cases and linking them to basic theoretical properties. The proposed general viewpoint and method can be interesting to facilitate the design of controllers achieving coordinated motion in new settings.

A particular application which has attracted much attention is the synchronization of rigid body attitudes, often as an abstraction of satellites, see e.g. [70, 72, 86, 89, 98, 109, 110, 135, 147]. In most of these papers, the synchronization algorithm actually depends on tracking a common reference or a leader which is imposed to the agents, such that the actual agreement problem is circumvented. The present dissertation contains a specific chapter about satellite attitude synchronization with mechanical dynamics, insisting on issues related to invariance and the agreement problem, in absence of a reference or leader. The relevance of such a setting is briefly discussed in the previous sections and in the corresponding Chapter 7.

During the last decades, the geometric viewpoint has become a well-established tool in analysis and control of *mechanical systems*, see e.g. [5, 60, 81, 83]. During the same time, a geometric approach to *algorithm design* has been successfully developed, see e.g. [18, 19, 38] and the books [2, 49]. The main concern in the present work differs from both approaches, because it focuses on fundamental problems related to the coupling of geometry and coordination rather than on complex mechanical dynamics (observability and controllability in underactuated settings, nonholonomic constraints,...) or optimized local convergence rates. However, its geometric mindset follows from these traditions, relayed through previous work in the laboratory at University of Liège.

The relation to existing literature is further discussed at the beginning of each part of the dissertation.

1.4 Summary and Outline of the presentation

The general spirit of the present dissertation is rather conceptual: it introduces and examines a bunch of simple ideas, associated to simplified problems, without necessarily going into all details relevant in more realistic applications; it tries to highlight general facts and phenomena associated to basic properties of the problem settings, and to propose general strategies

to address them; thanks to this conceptual viewpoint, it establishes links between several results that might a priori appear to be completely independent.

The main points of focus in this dissertation are amply introduced in the previous sections:

- defining *coordinated behaviors* of a swarm of agents — consisting of specific configurations and collective motions — and designing/analyzing control laws to reach them,
- on *nonlinear manifolds* — like the circle, sphere, rotation matrices or rigid body motions — in contrast to vector spaces,
- in a *decentralized, autonomous-agent setting* where each agent controls its motion individually, in order to achieve its individual objective in the best possible way, without being told what to do by any potential "supervising coordinator",
- maintaining the *symmetry / invariance* of the setting, in such a way that the swarm's behavior is independent of the absolute position of the swarm in configuration space, and coordination involves an *agreement problem* among agents of the swarm.

Relevant points that *are not addressed* in the present work are the influence of *time delays* that are unavoidable in practice, as well as the behavior of swarms with *state-dependent interconnection graphs*, i.e. where the possibility of communication between two agents depends on their relative positions (on the original manifold where coordination is to be achieved or in a higher-dimensional state space). The latter problem would require to analyze not only how the system behaves under specific assumptions on the interconnection graph, as is done in the present dissertation, but also how the graph can evolve as a consequence of the motions of the agents; a brief discussion about this essentially open problem is given at the end of Chapter 3.

The following briefly summarizes the further content of the dissertation and its organization. In addition to the introductory chapters and conclusion, the dissertation is subdivided into three parts that can be read more or less independently. Each part is introduced by a more detailed description of its content, along with an associated literature review, and concluded by a recapitulation of the main results where original contributions are highlighted. Illustrative examples are distributed throughout the chapters to facilitate understanding of proposed concepts. A convergence analysis is provided for all the proposed algorithms. Some open questions are formulated along the way — tagged as **O.Q.** — to be summarized again in the overall conclusion.

• Before presenting the core of the work, a chapter about mathematical background defines recurrent notation and collects some specific tools that are important ingredients in the sequel. This includes a brief review of graph theoretic concepts, some results that are recurrently used to prove convergence of dynamical systems, and an introduction along with some specific properties about Lie groups and manifolds.

• The first chapter of Part I establishes the connection from the consensus problem on vector spaces, extensively studied in the existing literature, to *synchronization algorithms on the circle*. It analyzes the convergence properties of such algorithms on the circle and thereby illustrates several specific difficulties encountered for synchronization on nonlinear manifolds. A second chapter then proposes several alternative strategies to enhance synchronization properties on the circle: changing the interaction profile between agents; using a "gossip algorithm" in which agents randomly select a single neighbor; and exchanging additional auxiliary variables among agents to "cheat" the nonconvexity of the manifold.

• Part II generalizes the configuration space from the circle to more general *connected compact homogeneous manifolds*, like the sphere or the group of rotation matrices, but is still concerned about achieving *specific relative positions* of the agents. A first chapter starts by defining a *mean position* of points on such manifolds that is easily computable, calling it the induced arithmetic mean. On this basis, considering agents which would try to get as close as possible or as far as possible from their neighbors, it defines particular sets of relative positions among agents in the swarm, called *(anti-)consensus configurations*; in particular, balancing is defined as a configuration where the induced arithmetic mean of the agents' positions is the whole manifold. In a second chapter, gradient algorithms are designed to drive the individual agents towards such configurations, generalizing the gradient synchronization algorithms of the circle. An algorithm with auxiliary variables is also proposed to attain synchronization or balancing under weak conditions on the links between agents; other alternatives are briefly examined. A third chapter concludes the part with an application to the problem of rigid body attitude synchronization, considering a fully actuated torque-controlled mechanical model. The relation between the geometric approach and existing work on the topic is established, and alternative control laws are proposed. Extensions include the use of auxiliary variables to obtain attitude synchronization under weak conditions on the links between agents, as well as a local attitude synchronization algorithm that leaves the swarm free to move like a rigid body once synchronized.

• Part III turns to *coordinated motion*, which is the second type of coordination problem identified in the Introduction. The initial setting is a swarm of agents moving on a *Lie group*. A first chapter defines relative positions from basic symmetry principles. Then it defines two types of coordinated motion, called "left-invariant" and "right-invariant" coordination, as motions with fixed relative positions. It shows that left-invariant (respectively right-invariant) coordination is equivalent to equal right-invariant (respectively left-invariant) velocities in the associated Lie algebra. Requiring both types of coordination simultaneously leads to a third type of coordinated motion called "total coordination". Given a velocity, total coordination is only feasible for specific "compatible" relative positions of the agents; this relationship is closely examined and involves several classical objects of group theory. To conclude the chapter, it is shown that the theoretical concepts have a clear intuitive meaning for the groups of rigid body motion. In a second chapter, a methodology is proposed for systematic design of control algorithms achieving coordinated motion. Right-invariant or fully actuated left-invariant coordination can be solved simply by a vector space consensus algorithm. Total coordination requires a more complex methodology in order to control the relative positions towards compatible positions. The same kind of strategy is necessary to obtain left-invariant coordination with underactuated agents. Finally, it is shown how to combine this framework for coordinated motion with the requirement to form and maintain specific configurations, as defined in Parts I and II of the dissertation.

• The overall conclusion first summarizes the main contributions in a different way, extracting results from different chapters and sections in order to highlight related observations. After this, it briefly elaborates on the distance from this rather conceptual dissertation to practical applications. Then it indicates several open questions and more general directions for related future research. It concludes with some general messages suggested by this work.

1.5 Publications

The results of the present work are published in the following papers.

A. Sarlette, R. Sepulchre and N.E. Leonard, "Discrete-time synchronization on the N-torus", *Proc. 17th Intern. Symp. Math. Theory of Networks and Systems*, pp. 2408–2414, 2006.

L. Scardovi, A. Sarlette and R. Sepulchre, "Synchronization and balancing on the N-torus", *Systems and Control Letters* 56(5), pp. 335–341, 2007.

A. Sarlette, S.E. Tuna, V.D. Blondel and R. Sepulchre, "Global synchronization on the circle", *Proc. 17th IFAC World Congress*, 2008.

A. Sarlette and R. Sepulchre, "Consensus optimization on manifolds", to be published in *SIAM J. Control and Optimization*, 2008.

A. Sarlette, R. Sepulchre and N.E. Leonard, "Cooperative attitude synchronization in satellite swarms: a consensus approach", *Proc. 17th IFAC Symp. Automatic Control in Aerospace*, 2007.

A. Sarlette, R. Sepulchre and N.E. Leonard, "Autonomous rigid body attitude synchronization", *Proc. 46th IEEE Conf. Decision and Control*, pp. 2566–2571, 2007.

A. Sarlette, R. Sepulchre and N.E. Leonard, "Autonomous rigid body attitude synchronization", to be published in *Automatica*, 2008.

A. Sarlette, S. Bonnabel and R. Sepulchre, "Coordination on Lie groups", *Proc. 47th IEEE Conf. Decision and Control*, pp. 1275–1279, 2008.

A. Sarlette, S. Bonnabel and R. Sepulchre, "Coordinated motion design on Lie groups", submitted to *IEEE Trans. Automatic Control*, 2008.

Chapter 2

Mathematical preliminaries

2.1 General notation and definitions

In addition to the specific notation introduced in the remainder of this chapter and in the core of the dissertation, the following general notation is used.

The sequence of integers from 1 to $N > 1$ is summarized by notation $1, 2, ..., N$. Similarly, the set of elements a_k for index k running from 1 to $N > 1$ is summarized by $a_1, a_2, ..., a_N$. Notation $x \in E$ (respectively $x \notin E$) means that element x belongs to (respectively does not belong to) the set E. The set of elements x belonging to some set $\mathcal{X}$ and satisfying conditions $c_1, c_2, ..., c_N$ is denoted $\{x \in \mathcal{X} : c_1, c_2, ..., c_N\}$. The fact that a set F is a subset of a set E is denoted $F \subset E$, or $F \subseteq E$ if F may also be equal to E. Given two sets E and F, their union and intersection are denoted $E \cup F$ and $E \cap F$ respectively. The union — resp. intersection — of sets $E_1, E_2, ..., E_N$ is summarized by $\cup_k E_k := \{x : x$ belongs to at least one of the E_k, for $k = 1, 2, ..., N\}$ — resp. by $\cap_k E_k := \{x : x$ belongs to every E_k, for $k = 1, 2, ..., N\}$. The difference between sets E and F is denoted $E \setminus F = \{x : x \in E$ and $x \notin F\}$. Given a finite set E, the number of elements in E is denoted $|E|$.

The sum of elements $a_1, a_2, ..., a_N$ is denoted $\sum_{k=1}^{N} a_k$. If the indices in the sum must satisfy an additional condition, it is added to the index of the summation symbol $\sum$. Sometimes, when it is clear from the context, the range of the summation index k is not explicitly noted.

The set of all integers is denoted $\mathbb{Z}$. The set of all real numbers is denoted $\mathbb{R}$ and the set of all complex numbers is denoted $\mathbb{C}$. The unit imaginary number is denoted $i = \sqrt{-1}$. A complex number $x \in \mathbb{C}$ can be written as a sum of its real and imaginary parts, $x = \Re e(x) + i\,\Im m(x)$, or using its norm and argument, $x = \|x\|\, e^{i \arg(x)}$; for $x = 0$, the argument is not uniquely defined and the present work takes the convention that it can take any value on the circle S^1. The set of positive reals and of non-negative reals are denoted $\mathbb{R}_{>0}$ and $\mathbb{R}_{\geq 0}$ respectively; similar indexing can be applied with $<$ and $\leq$, with respect to other references than 0, as well as to other sets, like $\mathbb{Z}$. For $x \in \mathbb{R}$, notation $\lfloor x \rfloor$ denotes the "integer part" of x, that is the largest integer $m \in \mathbb{Z}$ such that $m \leq x$. Notation $a \gg b$ denotes that a is much larger than b, in the sense that $\frac{a}{b}$ is several orders of magnitude larger than 1. Notation $b = \mathcal{O}(a)$ denotes a quantity b "of the order of" a, in the sense that $\frac{a}{b} \simeq 1$. This notation is mostly used to characterize the evolution of quantities $a(t)$ and $b(t)$ when t tends towards a

particular value: then $b = \mathcal{O}(a)$ means that $\frac{a(t)}{b(t)}$ tends to a finite limit when t tends towards the particular value of interest (often 0 or infinity).

The set of all vectors containing n real elements is denoted $\mathbb{R}^n$. The set of all real matrices of n rows and m columns is denoted $\mathbb{R}^{n\times m}$. Similar definitions can be made with $\mathbb{R}$ replaced by some other set A. Notation $E \times F$ denotes the cartesian product of two sets E and F, for instance $\mathbb{R}^2 = \mathbb{R} \times \mathbb{R}$. By default, the element in row j and column k of matrix A is denoted a_{jk}. Similarly, elements a_{jk} are canonically collected in a corresponding matrix A. The vector of $\mathbb{R}^n$ whose elements are all equal to 1 is denoted $\mathbf{1}_n$. The identity matrix in $\mathbb{R}^{n\times n}$ is denoted I_n. Notation $\text{diag}(x)$, with $x \in \mathbb{R}^n$, denotes the square matrix whose diagonal elements are the elements of x and whose off-diagonal elements are all zero. The determinant of $A \in \mathbb{R}^{n\times n}$ is denoted $\det(A)$; its trace is denoted $\text{trace}(A)$. The transpose of $A \in \mathbb{R}^{m\times n}$ is denoted A^T, and its inverse (for $m = n$ and $\det(A) \neq 0$) is denoted A^{-1}. The Kronecker product of two matrices A and B is denoted $A \otimes B$. The set of $n \times n$ symmetric positive semidefinite matrices is denoted $\mathbb{S}_n^+$.

A square matrix $B \in \mathbb{R}^{n\times n}$ can be decomposed into UR with U orthogonal and R symmetric positive semidefinite; R is always unique, U is unique if B is non-singular (see [20]).

The absolute value of $x \in \mathbb{R}$ is $|x|$. For a vector, matrix or higher-order tensor x, the same notation $|x|$ denotes the tensor whose elements are the absolute values of the elements of x. For a vector $x \in \mathrm{R}^n$, $\|x\|$ denotes its Euclidean norm, i.e. $\|x\| = \sqrt{x^T x} = \sqrt{\sum_{k=1}^n x_k^2}$. For a matrix $x \in \mathbb{R}^{m\times n}$, the canonical norm used in this dissertation is the Frobenius norm $\|x\| = \sqrt{\text{trace}(x^T x)} = \sqrt{\sum_{k=1}^n \sum_{j=1}^m x_{jk}^2}$.

A function associating an element of set $\mathcal{Y}$ to each element of a set $dom(f) \subseteq \mathcal{X}$ is denoted $f(x) : \mathcal{X} \to \mathcal{Y}$. $\mathcal{X}$ can be larger than the domain of f, the latter consisting of the restricted set $dom(f)$ on which the function is applied. Consider a real function $f(x) : \mathcal{X} \to \mathbb{R}$ of a variable x evolving in an arbitrary space $\mathcal{X}$. The maximum value of $f(x)$ over a domain $B \subset \mathcal{X}$ is denoted $\max_{x\in B}(f(x))$. The value of x achieving this maximum is denoted $\text{argmax}_{x\in B}(f(x))$. A similar notation is used with min and argmin for the minimum. Sometimes, the domain B is not explicitly specified when it is clear from the context.

For a differentiable function $f(x) : \mathbb{R} \to \mathbb{R}$, the derivative of f with respect to x is written $\frac{d}{dx}f$; if the latter is to be evaluated at a specific point x_0, this is written $\frac{d}{dx}f|_{x_0}$. For a differentiable function $f(x) : \mathbb{R}^n \to \mathbb{R}$, the derivative of f with respect to the kth component of $x = (x_1, x_2, ..., x_n)$ is denoted $\frac{\partial f}{\partial x_k}$ or $\text{grad}_k(f)$. The vector with components $\left(\frac{\partial f}{\partial x_1}, \frac{\partial f}{\partial x_2}, ..., \frac{\partial f}{\partial x_n}\right)$ is denoted $\frac{\partial f}{\partial x}$ or $\text{grad}(f)$ (see Section 2.4 for a definition of the gradient and the associated notation). With some abuse, the same notation is also used with x_k multidimensional; for instance, if $x \in \mathbb{R}^8$ is actually — for reasons clear from the context — the collection of four 2-dimensional vectors x_1, x_2, x_3, x_4, then $\frac{\partial f}{\partial x_1}$ would be a two-dimensional vector containing the derivatives of f with respect to the two components of $x_1 \in \mathbb{R}^2$. A *critical point* of a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ is a point where $\frac{\partial f}{\partial x_k} = 0$ for $k = 1, 2, ..., n$. A function is called *smooth* if it is infinitely differentiable.

The vector product between $x_1 \in \mathbb{R}^3$ and $x_2 \in \mathbb{R}^3$ is denoted $x_1 \wedge x_2 \in \mathbb{R}^3$. Notation $x!$ for $x \in \mathbb{Z}_{>0}$ denotes the product $1 \cdot 2 \cdot 3 \cdot ... \cdot x$.

2.2 Fundamentals of graph theory

In the framework of coordination with limited agent interconnections, it is customary to represent communication links by means of a *graph*. The present section contains some fundamental elements of graph theory, focusing on various definitions of *connectivity* and related properties of a particular matrix called the *graph Laplacian*. Unless explicitly mentioned, this material is standard and can be found in any textbook on the topic, like for instance [27, 36].

2.2.1 Representing a network of interconnected agents by a graph

Definition 2.2.1: *A* directed graph $\mathbb{G}(\mathcal{V}, \mathcal{E})$ *(short* digraph $\mathbb{G}$*) is composed of a finite set* $\mathcal{V}$ *of* vertices*, and a set* $\mathcal{E}$ *of* edges *which are ordered pairs of vertices* (j, k) *with* j *and* $k \in \mathcal{V}$*.*

In the present work, each agent is identified with a vertex; the N agents = vertices are designed by positive integers $1, 2, ..., N$, so $\mathcal{V} = \{1, 2, ..., N\}$. Edges represent interconnections among the vertices: the presence of edge (j, k) has the meaning that agent j sends information to agent k, or equivalently, agent k measures quantities concerning agent j. The present work assumes that no edge is needed for an agent k to get information about itself; it therefore uses the convention $(k, k) \notin \mathcal{E}\ \forall k \in \mathcal{V}$, i.e. $\mathbb{G}$ contains no *self-loops*. In visual representations of a graph, a vertex is depicted by a point, and edge (j, k) by an arrow from j to k (see Figure 2.1.a). Therefore a frequent alternative notation for $(j, k) \in \mathcal{E}$ is $j \rightsquigarrow k$. Introducing more vocabulary, one says that j is an *in-neighbor* of k and k is an *out-neighbor* of j.

Definition 2.2.2: *An* undirected graph $\mathbb{G}(\mathcal{V}, \mathcal{E})$ *(short* graph $\mathbb{G}$*) is a digraph in which* $(k, j) \in \mathcal{E}$ *whenever* $(j, k) \in \mathcal{E}$*, for all* j*,* $k \in \mathcal{V}$*.*

Equivalently, an undirected graph can be defined as a set of vertices and a set of *unordered* pairs of vertices. In the visual representation of an undirected graph, all arrows are bidirectional; therefore arrowheads are usually dropped (see Figure 2.1.b). One simply says that j and k are *neighbors* and writes $j \sim k$ instead of $j \rightsquigarrow k$ and $k \rightsquigarrow j$.

Figure 2.1: (a) A directed graph on 5 vertices. (b) An undirected graph on 5 vertices.

2.2.2 Weighted graphs, algebraic representation and Laplacian

Definition 2.2.3: *A* weighted digraph $\mathbb{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ *is a digraph associated with a set* $\mathcal{A}$ *that assigns a positive* weight $a_{jk} \in \mathbb{R}_{>0}$ *to each edge* $(j, k) \in \mathcal{E}$*.*

In contrast, graphs without weights are also called *unweighted*, or *unit-weighted*. The latter denomination comes from the following fact: a weighted graph can be defined by its vertices and weights only, by extending the weight set to all pairs of vertices and imposing $a_{jk} = 0$ if and only if (j, k) does not belong to the edges of $\mathbb{G}$; then an unweighted graph has

the binary representation

$$a_{jk} = 1 \quad \text{if } (j,k) \text{ is an edge of } \mathbb{G}\,, \qquad a_{jk} = 0 \quad \text{otherwise.} \tag{2.1}$$

A weighted digraph is said to be *undirected* if $a_{jk} = a_{kj}\ \forall j,k \in \mathcal{V}$. It may happen that $(j,k) \in \mathcal{E}$ whenever $(k,j) \in \mathcal{E}\ \forall j,k \in \mathcal{V}$, but $a_{jk} \neq a_{kj}$ for some $j,k \in \mathcal{V}$; in this case the graph is called *bidirectional*.

Definition 2.2.4: *The* in-degree *of vertex k is $d_k^{(i)} = \sum_{j=1}^{N} a_{jk}$. The* out-degree *of vertex k is $d_k^{(o)} = \sum_{j=1}^{N} a_{kj}$.*

For unweighted digraphs, the in- and out-degree give the number of in- and out-neighbors respectively.

Definition 2.2.5: *A digraph is said to be* balanced *if $d_k^{(i)} = d_k^{(o)}\ \forall k \in \mathcal{V}$.*

In particular, undirected graphs are balanced.

There are several equivalent ways to represent a graph $\mathbb{G}$ by a matrix of finite dimensions. The most natural way is to define the *adjacency matrix* $A \in \mathbb{R}^{N\times N}$ which contains a_{jk} in row j, column k; adjacency matrix A is symmetric if and only if $\mathbb{G}$ is undirected. An alternative way often encountered in the literature is the *incidence matrix* $B \in (\{-1,0,1\})^{N\times|\mathcal{E}|}$. For a digraph $\mathbb{G}$, each column of B corresponds to one edge and each row to one vertex; if column m corresponds to edge (j,k), then

$$b_{jm} = -1\,, \qquad b_{km} = 1 \qquad \text{and} \qquad b_{lm} = 0 \text{ for } l \notin \{j,k\}\,.$$

For an undirected graph $\mathbb{G}$, each column corresponds to an undirected edge; an *arbitrary orientation* (j,k) or (k,j) is chosen for each edge and B is built for the resulting directed graph. Thus B is not unique for a given $\mathbb{G}$, but $\mathbb{G}$ is unique for a given B.

The in- and out-degrees of vertices $1,2,...,N$ can be assembled in diagonal matrices $D^{(i)}$ and $D^{(o)}$; when $D^{(o)} = D^{(i)}$ it is often simply noted D. Since the adjacency matrix A has zero diagonal elements with the convention of the present work, no information is lost by taking linear combinations of A with a diagonal matrix. The *Laplacian matrix* associated to $\mathbb{G}$ is particularly interesting.

Definition 2.2.6: *The* in-Laplacian *of a digraph $\mathbb{G}$ is defined as $L^{(i)} = D^{(i)} - A$. Similarly, the associated* out-Laplacian *is $L^{(o)} = D^{(o)} - A$. For a balanced graph $\mathbb{G}$, the* Laplacian *$L = L^{(i)} = L^{(o)}$.*

Although Definition 2.2.6 extends it to the weaker condition of a balanced graph, the standard definition of Laplacian L is for undirected graphs. For the latter, L is symmetric and, remarkably, $L = BB^T$ where B is the incidence matrix. For general digraphs, by construction, $(\mathbf{1}_N)^T L^{(i)} = 0$ and $L^{(o)}\,\mathbf{1}_N = 0$. The spectrum of the Laplacian reflects several interesting properties of the associated graph, specially in the case of undirected graphs, see for example [27]. In particular, it reflects its *connectivity* properties.

2.2.3 Graph connectivity and time-varying graphs

A *directed path* of length l from vertex j to vertex k is a sequence of vertices $v_0, v_1, ..., v_l$ with $v_0 = j$ and $v_l = k$ and such that $(v_m, v_{m+1}) \in \mathcal{E}$ for $m = 0,1,...,l-1$. An *undirected path*

between vertices j and k is a sequence of vertices $v_0, v_1, ..., v_l$ with $v_0 = j$ and $v_l = k$ and such that $(v_m, v_{m+1}) \in \mathcal{E}$ or $(v_{m+1}, v_m) \in \mathcal{E}$, for $m = 0, 1, ..., l-1$. The following connectivity properties are of decreasing strength.

Definition 2.2.7: *A digraph $\mathbb{G}$ is* strongly connected *if it contains a directed path from every vertex to every other vertex (and thus also back to itself). A digraph $\mathbb{G}$ is* root-connected *if it contains a node k, called the* root*, from which there is a directed path to every other vertex (but not necessarily back). A digraph $\mathbb{G}$ is* weakly connected *if it contains an undirected path between every two of its vertices.*

For an undirected graph $\mathbb{G}$, all these notions become equivalent and are simply summarized by the term *connected*.

Definition 2.2.8: *An undirected graph $\mathbb{G}$ is* connected *if it contains an undirected path between every two of its vertices.*

When an undirected graph is not connected (resp. a digraph is not weakly connected), it can be partitioned into several *connected components* (resp. weakly connected components).

Definition 2.2.9: *A* connected component *of a graph $\mathbb{G}$ is a subset $\mathcal{V}_c \subseteq \mathcal{V}$ of vertices such that $\mathbb{G}$ contains a path between every two vertices of $\mathcal{V}_c$, but no edge between any vertex of $\mathcal{V} \setminus \mathcal{V}_c$ and a vertex of $\mathcal{V}_c$.*

For $\mathbb{G}$ representing interconnections in a network of agents, clearly coordination can only take place if $\mathbb{G}$ is connected. If this is not the case, coordination will only be achievable separately in each connected component of $\mathbb{G}$. A more interesting discussion of connectivity arises when the graph $\mathbb{G}$ varies with time. Before discussing this case, the following summarizes some spectral properties of the Laplacian that are linked to the connectivity of the associated graph.

Properties 2.2.10 (Laplacian): *The out-Laplacian $L^{(o)}$ of a digraph $\mathbb{G}$ has the following properties.*

(a) All eigenvalues of $L^{(o)}$ have nonnegative real parts.
(b) If $\mathbb{G}$ is strongly connected, then 0 is a simple eigenvalue of $L^{(o)}$.
(c) The quadratic expression $x^T L x$, with $x \in \mathbb{R}^N$, is positive semidefinite if and only if $\mathbb{G}$ is balanced.

The Laplacian L of an undirected graph $\mathbb{G}$ has the following properties.

(d) L is symmetric positive semidefinite.
(e) The algebraic and geometric multiplicity of 0 as an eigenvalue of L is equal to the number of connected components in $\mathbb{G}$.

The properties of L for an undirected graph are much stronger than those of $L^{(o)}$ for a digraph. In particular, the multiplicity of 0 as an eigenvalue of $L^{(o)}$ does not indicate the number of weakly connected components in $\mathbb{G}$. In the sequel, spectral properties of the Laplacian will mostly be useful only for undirected graphs (sometimes also for balanced graphs).

In a coordination problem, interconnections among agents can vary with time as some links are dropped and others are established. In this case, the communication links are represented by a *time-varying graph* $\mathbb{G}(t)$ in which the vertex set $\mathcal{V}$ is fixed (by convention), but edges $\mathcal{E}$ and weights $\mathcal{A}$ can depend on time. All the previous definitions carry over to time-varying graphs; simply, each quantity depends on time. The following definition prevents edges from

vanishing or growing indefinitely.

Definition 2.2.11: *A time-varying graph $\mathbb{G}(t)$ is a* δ-digraph *if the elements of matrix $A(t)$ are bounded and satisfy the threshold $a_{jk}(t) \geq \delta > 0 \ \forall (j,k) \in \mathcal{E}(t)$, for all t.*

The present work always assumes that time-varying graphs are δ-digraphs; in addition, in continuous-time they are assumed to be piecewise continuous. For δ-digraphs $\mathbb{G}(t)$, it is intuitively clear that coordination may be achieved if information exchange is "sufficiently frequent", without requiring it to take place all the time. The following definitions of "integrated connectivity over time" can be found in [13, 94, 95, 143].

Definition 2.2.12: (from [94, 95]) *In discrete-time, for a δ-digraph $\mathbb{G}(\mathcal{V}, \mathcal{E}(t), \mathcal{A}(t))$ and some constant $T \in \mathbb{Z}_{\geq 0}$, define the graph $\bar{\mathbb{G}}(\mathcal{V}, \bar{\mathcal{E}}(t), \bar{\mathcal{A}}(t))$ where $\bar{\mathcal{E}}(t)$ contains all edges that appear in $\mathbb{G}(\tau)$ for $\tau \in [t, t+T]$ and $\bar{a}_{jk}(t) = \sum_{\tau=t}^{t+T} a_{jk}(\tau)$. Similarly, in continuous-time, for a δ-digraph $\mathbb{G}(\mathcal{V}, \mathcal{E}(t), \mathcal{A}(t))$ and some constant $T \in \mathbb{R}_{>0}$, define the graph $\bar{\mathbb{G}}(\mathcal{V}, \bar{\mathcal{E}}(t), \bar{\mathcal{A}}(t))$ by*

$$\bar{a}_{jk}(t) = \begin{cases} \int_t^{t+T} a_{jk}(\tau) d\tau & \text{if } \int_t^{t+T} a_{jk}(\tau) d\tau \geq \delta \\ 0 & \text{if } \int_t^{t+T} a_{jk}(\tau) d\tau < \delta \end{cases}$$

$$(j,k) \in \bar{\mathcal{E}}(t) \text{ if and only if } \bar{a}_{jk}(t) \neq 0 \ .$$

Then $\mathbb{G}(t)$ is said to be uniformly connected over T *if there exists a time horizon T and a vertex $k \in \mathcal{V}$ such that $\bar{\mathbb{G}}(t)$ is root-connected with root k for all t.*

2.2.4 Some particular graphs

Some particular types of graphs are favorable for coordination, others are handy to provide counterexamples. The following graphs are regularly used in the present dissertation.

Definition 2.2.13: *The (equally weighted)* complete graph *on N vertices is an unweighted, undirected graph that contains an edge between any pair of vertices.* (see Figure 2.2.a)

The complete graph on N vertices has $\frac{N(N-1)}{2}$ undirected edges.

Definition 2.2.14: *An* undirected ring *or* cycle graph *on $N > 1$ vertices is equivalent to an undirected path containing all vertices, to which is added an edge between the extreme vertices of the path. Similarly, a* directed ring *or* cycle graph *on $N > 1$ vertices is equivalent to a directed path containing all vertices, to which is added an edge from the last to the first vertex in the path.* (see Figures 2.2.b and 2.2.c)

For an undirected graph on 2 vertices, there are just two choices: either connect them or not; the connected graph is the complete graph and a cycle on 2 vertices. For $N > 2$, the undirected cycle on N vertices has N undirected edges. For $N > 1$, the directed cycle on N vertices has N directed edges.

Definition 2.2.15: *An* undirected tree *is a connected undirected graph in which it is impossible to select a subset of at least 3 vertices and a subset of edges among them to form an undirected cycle. A* directed tree *of root k is a root-connected digraph of root k, in which every vertex can be reached from k by following one and only one directed path.* (see Figures 2.2.d and 2.2.e)

A tree (directed or undirected) on N vertices has $N-1$ (directed or undirected) edges. In particular, a directed tree contains no cycle. In a directed tree $\mathbb{G}$, the (unique) in-neighbor of

a vertex j is called its *parent* and its out-neighbors are its *children*. The root has no parent, and the vertices with no children are called the *leaves*. The *descendants* of a vertex j are all the vertices that can be reached by following a directed path starting at j. The *ancestors* of a vertex j are all the vertices of which j is a descendant. These definitions can be carried over to an undirected graph after selecting an arbitrary root $k \in \mathcal{V}$ and then (uniquely) associating directions to the edges in order to build a directed tree of root k.

Definition 2.2.16: *A* (directed) spanning tree *of (di)graph* $\mathbb{G}(\mathcal{V}, \mathcal{E}, \mathcal{A})$ *is a (directed) tree on all vertices of* $\mathcal{V}$ *and whose edges are in* $\mathcal{E}$. (see Figure 2.2.f)

According to Definition 2.2.7, a digraph is root-connected if and only if it has a directed spanning tree; it is strongly connected if and only if has a directed spanning tree of root k for each $k \in \mathcal{V}$. Similarly, Definition 2.2.8 implies that an undirected graph is connected if and only if it has a spanning tree.

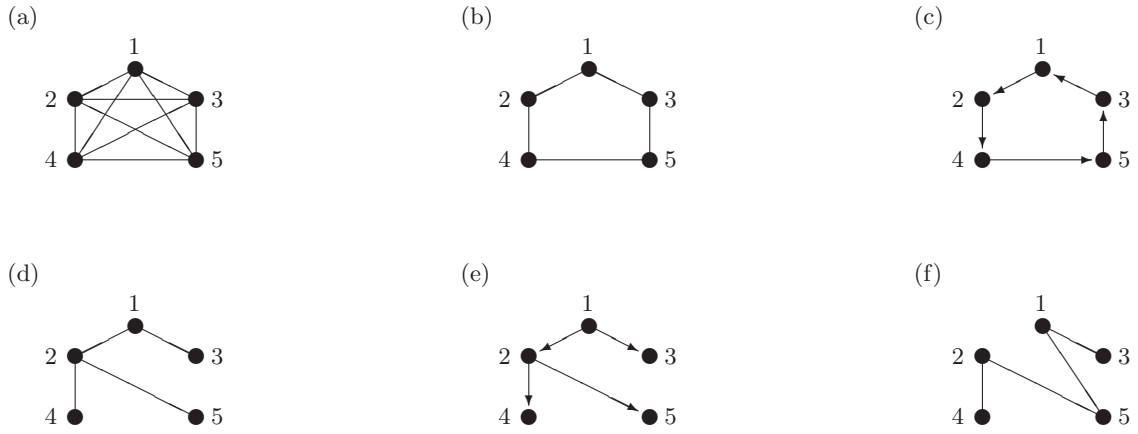


Figure 2.2: Graphs on 5 vertices: (a) complete graph. (b) undirected ring. (c) directed ring. (d) undirected tree. (e) directed tree of root 1. (f) a spanning tree of the graph in Figure 2.1.b ; (d) is also a spanning tree of that graph.

2.3 Convergence results used in the proofs

The present section reviews a set of results used in the convergence proofs in this dissertation. The goal is to make all the proofs in Parts I, II and III understandable for the reader who is not too familiar with nonlinear systems theory. Sometimes the results are not provided in their full generality, but convey the general idea needed in the proofs. For instance, all results are presented in their classical form for systems on a vector space $\mathbb{R}^n$, while the present dissertation uses them on nonlinear manifolds; it is believed that extensions like transposing the results from $\mathbb{R}^n$ to manifolds will be clear enough in the appropriate context, although presenting the most general results at the present place would require too much introduction and complicate notations. Most of the presented properties are classical results of nonlinear systems theory and can be found in reference books like [66, 128]. The results about asymptotically autonomous systems and chain recurrent sets are less common; references are

correspondingly identified. The reader is referred to the references for the proofs.

The general framework for the following results is a continuous-time dynamical system of the form

$$\frac{d}{dt}x = f(x,t) \tag{2.2}$$

where $x \in \mathbb{R}^n$ is the *state* of the system, $t \in \mathbb{R}$ is the time, and f is a function from $\mathbb{R}^n \times \mathbb{R}$ to $\mathbb{R}^n$ specifying the system's evolution. This is viewed in the usual sense that, starting from an initial condition (x_0, t_0), the evolution of x for $t > t_0$ is obtained by integrating (2.2) in time. Systems which are naturally modelled with higher-order time derivatives (like mechanical systems, where the position is derived twice) can be brought into form (2.2) by expanding the state x to include not only relevant "positions", but also some of their derivatives. The set on which positions evolve is then called the *configuration space*, whereas x belongs to the *state space*.

The function f must satisfy additional conditions for the system to be well-behaved. For instance, solutions of (2.2) starting at (x_0, t_0) are unique in the neighborhood of (x_0, t_0) where f is Lipschitz continuous. This condition is rather weak and will not be explicitly checked for all the algorithms proposed in the present dissertation.

Sometimes, a system displays a behavior A in the generic case, but there exist nasty initial conditions from which it does not behave like A, but rather like B. In such cases, it is sometimes possible to speak of a system with behavior A *for almost all initial conditions*. By this, it is meant that it displays behavior A for all possible initial conditions, except for a set of initial conditions that has measure zero with respect to all possible initial conditions.

If f in (2.2) does not depend on t, the system is called *autonomous*. The following always assumes autonomous systems, except for the results about asymptotically autonomous systems. An *equilibrium* of the system is a point in state space where $f(x) = 0$.

A major question to characterize systems like (2.2) is their behavior as t tends to infinity.

- A set S in state space is said to be *invariant* (resp. *positively invariant*) under (2.2) if all solutions starting in S at $t = 0$ remain in S for all $t \in \mathbb{R}$ (resp. for all $t \geq 0$).
- A set S in state space is said to be Lyapunov *stable* under (2.2) if for all $\varepsilon > 0$, there exists $\delta > 0$ such that all solutions starting in a *δ-neighborhood* of S — that is at a distance smaller than δ from S — remain in an ε-neighborhood of S for all future times.
- A solution of (2.2) is said to *asymptotically converge* towards S if the limit of the distance between the solution and set S for t tending to infinity is zero. It is said to *exponentially converge* if for $t > T$, the distance is smaller than $c\,e^{-\lambda t}$ for some $T > 0$, $c > 0$ and $\lambda > 0$.
- A set S in state space is said to be *asymptotically stable* under (2.2) if it is stable and there exists δ such that all solutions in a δ-neighborhood of S asymptotically converge to S.
- A set S in state space is said to be *(almost) globally asymptotically stable* if it is stable and the solutions starting at (almost) all possible initial conditions asymptotically converge to S. An asymptotically stable set that is not (almost) globally asymptotically stable is sometimes called *locally* asymptotically stable for emphasis.
- Denote by $\Phi(x_0, t)$ the point reached after time t by an autonomous system starting at point x_0. The positive *limit set* $L^+(x)$ of a point x is the set of points $\bar{x}$ in state space for which there exists an infinite sequence of instants $t_1, t_2, t_3, ...$ with t_k tending to $+\infty$

when k tends to $+\infty$ and such that $\Phi(x, t_k)$ tends to $\bar{x}$ when k tends to $+\infty$.

A mathematical result that is advantageously used to prove convergence of solutions of a system is Barbalat's Lemma.

Proposition 2.3.1: *(Barbalat's Lemma and Corollary) If the function $f(t) : \mathbb{R}_{\geq 0} \to \mathbb{R}^n$ is uniformly continuous and Riemann integrable over t ranging from 0 to $+\infty$, then the limit of f for t tending to $+\infty$ is zero. The same holds if $f(t)$ is uniformly continuous and $f(t) \in L_2(0, +\infty)$, i.e. the squared norm of $f(t)$ is integrable over t ranging from 0 to $+\infty$.*

Another popular tool for convergence analysis is *LaSalle's invariance principle*. Its main interest is that its approach can also be useful for controller design, as is extensively done in the present dissertation. The LaSalle invariance principle requires the existence of a *Lyapunov function* V : statespace $\to \mathbb{R}$ with the following properties:

- V is bounded below over all x in the state space;
- V is continuously differentiable;
- V never increases along solutions of the system, i.e. $\frac{d}{dt}V(x^*(t)) \leq 0 \ \forall t$ for every $x^*(t) : \mathbb{R} \to \mathbb{R}^n$ which is a solution of system (2.2) (with f independent of t).

Proposition 2.3.2: *(LaSalle's invariance principle) Consider a system (2.2) with f independent of t for which there exists a Lyapunov function satisfying the properties listed above. Denote by M the set of all points x in state space where $\frac{d}{dt}V(x) = 0$ when $\frac{d}{dt}x = f(x)$. Further assume that, for any possible initial condition x_0, there exists a* compact *set in state space inside which the solution of (2.2) starting at x_0 is ensured to stay. Then for any initial condition x_0, the corresponding solution $x(t)$ converges to the* largest invariant set in M *when t tends to infinity.*

The fact that the solution $x(t)$ stays in a compact set for any initial condition x_0 can be proved with several arguments. First, the state space may be naturally bounded (for instance, a circle or sphere), at least along some dimensions. Along remaining dimensions, the Lyapunov function V might be chosen *radially unbounded* in order to ensure that solutions cannot escape to infinity; that means, $V(x)$ must tend to infinity when any of the unbounded dimensions of x tends to infinity. Finally, symmetries or other conservation laws can ensure that a solution never leaves a subspace of the state space, thereby decomposing the state space along some dimensions into a foliation of separate trajectory sets. This is the set of arguments used in the proofs of the present dissertation; it will not be explicitly repeated in the proofs.

Consider a *controlled* system

$$\frac{d}{dt}x = f(x) + Bu \tag{2.3}$$

where $x \in \mathbb{R}^n$, signal $u(t) : \mathbb{R} \to \mathbb{R}^m$ is a control input that can be chosen and B is a matrix describing the range of the control term; without loss of generality, it is assumed that the columns of B form a set of orthonormal vectors. If $m < n$, the system is said to be *underactuated*. A different definition of underactuation is used in the framework of mechanical systems. For the latter, $x \in \mathbb{R}^n$ naturally splits into position variables $q \in \mathbb{R}^{n/2}$ and velocity variables $v \in \mathbb{R}^{n/2}$, with $n/2$ the number of *degrees of freedom* for the motion of the mechanical system. The part of (2.3) describing the evolution of q expresses the kinematic link between positions and velocities. The part of (2.3) describing the evolution of v expresses Newton's equation of motion and can be influenced by control forces u. Then a mechanical

system is said to be *underactuated* if $m < \frac{n}{2}$, i.e. if the dimension of the space spanned by control forces is smaller than the number of degrees of freedom of the system.

The notion of *controllability* is important for underactuated systems. Informally, a system like (2.3) is said to be *controllable* if any point in the state space can be reached by following a solution of (2.3) starting at any other point of the state space, by designing the control signal $u(t) : \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^m$ appropriately.

The following theorem gives an important convergence property for underactuated systems (with the first meaning, i.e. not mechanical systems) under a specific feedback control.

Proposition 2.3.3: *("Jurdjevic-Quinn theorem", adapted from [61]) Consider a system of the type (2.3) where in addition $f(x)$ is smooth. Assume that for $u = 0$, there exists a smooth Lyapunov function $V(x)$ satisfying the conditions for the LaSalle invariance principle. Further define $A(x) = \frac{\partial}{\partial y} f(y)|_{y=x} \in \mathbb{R}^{n \times n}$ and require that the matrix formed by column vectors B, $A(x)B$, $A(x)A(x)B$, $A(x)A(x)A(x)B$,... has rank n at each point x in the state space, except possibly at critical points of V. Then the feedback control*

$$u(x) = -B^T \frac{\partial}{\partial y} V(y)|_{y=x}$$

ensures that all solutions asymptotically converge to the set of critical points of V.

The present dissertation often involves controllers consisting of two "levels" that are interconnected in cascade. The following result basically ensures that the cascade of two *locally* asymptotically stable systems remains *locally* asymptotically stable. For most of the controllers proposed in the dissertation, this result is strengthened by a specific convergence proof.

Proposition 2.3.4: *([128, 136]) If S is an asymptotically stable set of the dynamical system $\frac{d}{dt}x = f(x)$ with state space $\subseteq \mathbb{R}^{m_1}$, and $y = 0$ is asymptotically stable for $\frac{d}{dt}y = g(y)$ with state space $\subseteq \mathbb{R}^{m_2}$, then S is a (locally) asymptotically stable set for the cascade system*

$$\begin{aligned} \tfrac{d}{dt}y &= g(y) \\ \tfrac{d}{dt}x &= f(x) + h(x, y) \end{aligned}$$

for any locally continuously differentiable function $h : \mathbb{R}^{m_1} \times \mathbb{R}^{m_2} \rightarrow \mathbb{R}^{m_1}$ satisfying $h(x, 0) = 0$ $\forall x \in$ state space $\subseteq \mathbb{R}^{m_1}$.

The local convergence result for "two-level" algorithms can often be improved thanks to the theory of asymptotically autonomous systems. A system (2.2) is *asymptotically autonomous* if f depends on t but there exists an autonomous *limit system* $f_p(x)$ such that, for each compact set S in the state space and each $\varepsilon > 0$, there is a time $T > 0$ for which $\|f(x,t) - f_p(x)\|^2 < \varepsilon$ for all $t > T$ and for all $x \in S$. The convergence properties of an asymptotically autonomous system can be characterized on the basis of the limiting autonomous system; this uses the notion of chain recurrent sets.

Definition 2.3.5: *Denote by $\Phi(x_0, t)$ the point reached after time t by an autonomous system starting at point x_0. A point $\bar{x}$ in the state space of the system is* chain recurrent *if and only if for every $\varepsilon > 0$, $T > 0$, there are sequences $\{x_0, x_1, x_2, x_3, ..., x_n\}$ of points in state space and $\{t_1, t_2, t_3, ..., t_n\}$ of instants in time with $x_n = x_0 = \bar{x}$ and such that*

- *$t_k \geq T$ for $k = 1, 2, ...n$;*
- *the distance between $\Phi(x_{k-1}, t_k)$ and x_k is smaller than ε for $k = 1, 2, ...n$.*

The chain recurrent set *of the system is the union of all its chain recurrent points.*

Proposition 2.3.6: (see [91]) *All solutions of an asymptotically autonomous system converge to the chain recurrent set of the limit system.*

Definition 2.3.5 of a chain recurrent set is difficult to use in practice. Fortunately, the following results allow an easier characterization of chain recurrent sets.

Proposition 2.3.7: *(see [107]) A point x in state space is chain recurrent if and only if it belongs to the intersection of all locally asymptotically stable sets containing the positive limit set $L^+(x)$ of x.*

Proposition 2.3.8: *(see [54]) Consider a system that can be written $\frac{d}{dt}x = \frac{\partial}{\partial y}V(y)|_{y=x}$ for some continuously differentiable function V :* statespace $\to \mathbb{R}$*; such a system is called a* gradient system*. If the state space has dimension n and V is at least n times continuously differentiable, then the chain recurrent set of the gradient system is equal to the set of its equilibria (which are the critical points of V).*

Proof: In [54], see Proposition 4 and Sard's theorem as explained just after it. □

2.4 Lie groups and homogeneous manifolds

The present dissertation considers coordination issues on nonlinear, but highly symmetric configuration spaces. In particular, synchronization and consensus is studied on *homogeneous manifolds* and coordinated motion is studied on *Lie groups*. Lie groups are a particular class of homogeneous manifolds with rich structure; examples include the set of rotation matrices and the set of rigid body motions. The sphere is an example of a homogeneous manifold that is not a Lie group.

The present section presents basic definitions and properties first of Lie groups, then of homogeneous manifolds. It briefly reviews how tangent spaces and gradient vectors are defined for manifolds embedded in a vector space $\mathbb{R}^m$. It finally introduces particular examples of Lie groups and homogeneous manifolds that often appear in practice and are used for illustration purposes in the dissertation: the circle, the group of rotation matrices $SO(n)$, the group of rigid body motions $SE(n)$ and the Grassmann manifolds $Grass(p, n)$.

Unless otherwise mentioned, this material is standard and can be found in any textbook on the topic. For Lie groups, the present summary uses notation close to [5]. The concepts of tangent vector, distance measure, Riemannian connections and gradients on manifolds are defined in a formal, intrinsic manner by the mathematical subject of Differential Geometry; see [25, 26] for a thorough but accessible mathematical treatment, and [2] for a summarized treatment intended for applied scientists. However, up to a few remarks, the present dissertation can be fully understood on the basis of the more traditional theory for embedded manifolds. Therefore concepts are introduced in this simplified setting.

Most of the convergence properties recalled in Section 2.3 for systems on $\mathbb{R}^n$ admit adaptations for systems on manifolds. In particular, the propositions involving Lyapunov functions or gradient systems are extended to manifolds; these extensions are essentially obtained by adapting the (meaning of the) notation and are not explicitly repeated in this dissertation. A particular adaptation of the Jurdjevic-Quinn theorem to Lie groups is presented and used in Part III.

2.4.1 Lie groups

Roughly speaking, a group G is a class of objects for which a "composition law" is defined, which associates to two objects g_1 and $g_2 \in G$ a third object $g_3 = g_1 g_2 \in G$. Typically, groups can be associated to "space transformation operators": combining two transformations of one class, another transformation of the same class is obtained — think for instance of the rotation transformation: combining two consecutive rotations is equivalent to applying one particular rotation. The composition law need not be commutative — applying rotation g_1 followed by rotation g_2 does not give the same result as applying rotation g_2 followed by rotation g_1 — but must satisfy a number of conditions. A *Lie* group is a "continuous" group, whose elements form a manifold (in opposition to *discrete* groups like for instance the symmetry groups of crystals). The mathematical definition of a Lie group is as follows; the present dissertation only considers finite-dimensional Lie groups.

Definition 2.4.1: *A Lie group G of finite dimension n is an n-dimensional smooth manifold among whose elements is defined a* multiplication operation *or* composition operation *associating an element $g_1 g_2 \in G$ to two elements g_1 and $g_2 \in G$ with the following properties.*

- *Associativity: if g_1, g_2 and $g_3 \in G$, then $(g_1 g_2) g_3 = g_1 (g_2 g_3) =: g_1 g_2 g_3$.*
- *Identity: there exists an element $e \in G$, called the identity, such that $eg = ge = g$ $\forall g \in G$.*
- *Inverse: if $g \in G$, then G also contains an element, denoted g^{-1} and called the* inverse *of g, such that $g g^{-1} = g^{-1} g = e$.*
- *The group operations are compatible with the smooth manifold structure; in particular, multiplication and inversion are differentiable maps.*

It is easy to show that for all $g \in G$, the inverse g^{-1} is uniquely defined.

A group G is called *Abelian* if $g_1 g_2 = g_2 g_1 \ \forall \ g_1, g_2 \in G$; Abelian groups are usually much easier to study. The multiplication operation $g_1 g_2$ can be seen alternatively as *left-multiplication* of g_2 by g_1, i.e. $g_1 g_2 = L_{g_1}(g_2)$ with $L_g : G \to G$, or as *right-multiplication* of g_1 by g_2, i.e. $g_1 g_2 = R_{g_2}(g_1)$ with $R_g : G \to G$.

Definition 2.4.2: *A* Lie subgroup *G_1 of G is a submanifold of G which, associated with the multiplication operation of G, forms a group compatible with the smooth submanifold structure.*

G_1 is a *subgroup* of G if contains the identity e, as well as the inverse and products of all its elements. Then Cartan's theorem can be used to ensure that the remaining requirements for G_1 to be a *Lie subgroup* are satisfied.

Proposition 2.4.3: *("Cartan's Theorem", see e.g. [106]) A subgroup G_1 of a Lie group G is a* Lie subgroup *of G if it is closed under the topology of G.*

Consider a Lie group G and a trajectory $g(t)$ on G, parametrized by time t. The velocity $\frac{d}{dt} g(t)|_{t=\tau}$ belongs to the tangent plane $T_{g(\tau)}G$ to G at $g(\tau)$. Multiplying $g(t)$ by some constant $h \in G$ on the left, the velocity $\frac{d}{dt}(L_h g(t))|_{t=\tau} = L_{*\, hg(\tau) \leftarrow g(\tau)} \frac{d}{dt} g(t)|_{t=\tau} \in T_{hg(\tau)}G$ where $L_{*\, hg(\tau) \leftarrow g(\tau)} : T_{g(\tau)}G \to T_{hg(\tau)}G$ is an invertible linear application. Similarly, multiplying $g(t)$ by some constant $h \in G$ on the right, the velocity $\frac{d}{dt}(R_h g(t))|_{t=\tau} = R_{*\, g(\tau)h \leftarrow g(\tau)} \frac{d}{dt} g(t)|_{t=\tau}$ $\in T_{g(\tau)h}G$ defines the invertible linear application $R_{*\, g(\tau)h \leftarrow g(\tau)} : T_{g(\tau)}G \to T_{g(\tau)h}G$.

Properties 2.4.4 (L_* and R_*): *The applications L_* and R_* of a Lie group G satisfy the*

following properties for any g_1, g_2 and $g_3 \in G$.

(a) $L_{*\,g_1 \leftarrow g_1} = R_{*\,g_1 \leftarrow g_1} = \text{Identity}$

(b) $L_{*\,g_3 \leftarrow g_2} L_{*\,g_2 \leftarrow g_1} = L_{*\,g_3 \leftarrow g_1}$ *and* $R_{*\,g_3 \leftarrow g_2} R_{*\,g_2 \leftarrow g_1} = R_{*\,g_3 \leftarrow g_1}$

(c) $L_{*\,g_2 g_1 g_3 \leftarrow g_1 g_3}\, R_{*\,g_1 g_3 \leftarrow g_1} = R_{*\,g_2 g_1 g_3 \leftarrow g_2 g_1}\, L_{*\,g_2 g_1 \leftarrow g_1}$

When the tangent space on which the linear application operates is clear from the context, it is in fact possible to write L_{*h} and R_{*h} to mean $L_{*\,hg \leftarrow g}$ and $R_{*\,gh \leftarrow g}$ respectively; after the present introduction, the dissertation uses this convention.

A particular case of the above is to take $h = g(\tau)^{-1}$. Then $L_{*h} = L_{*g(\tau)^{-1}}$ and Properties 2.4.4 imply

$$\tfrac{d}{dt}(L_h g(t))|_{t=\tau} = L_{*\,g(\tau)^{-1} g(\tau) \leftarrow g(\tau)}\, \tfrac{d}{dt} g(t)|_{t=\tau} = L_{*\,e \leftarrow g(\tau)}\, \tfrac{d}{dt} g(t)|_{t=\tau} =: \xi^l$$

where ξ^l belongs to the tangent space T_eG to G at the identity. A similar construction can be made with $R_{*h} = R_{*g(\tau)^{-1}}$ to define $\xi^r \in T_eG$. Those two objects in T_eG are instrumental in the study of motion on Lie groups.

Definition 2.4.5: *The* left-invariant velocity $\xi^l \in T_eG$ *associated to* $g(t)$ *evolving on* G *is defined by* $\xi^l(\tau) = L_{*g(\tau)^{-1}}\, \frac{d}{dt} g(t)|_{t=\tau}$. *Similarly, the* right-invariant velocity $\xi^r \in T_eG$ *associated to* $g(t)$ *evolving on* G *is defined by* $\xi^r = R_{*g(\tau)^{-1}}\, \frac{d}{dt} g(t)|_{t=\tau}$.

The left-invariant (respectively right-invariant) velocity of g does not change if g is multiplied on the left (respectively on the right) by any constant $h \in G$. Although they belong to the same vector space T_eG, in general $\xi^r \neq \xi^l$ (see examples at the end of the present section). However, ξ^l and ξ^r are not independent.

Definition 2.4.6: *The* adjoint representation Ad_g *of* $g \in G$ *is the linear operator* $Ad_g = L_{*g} R_{*g^{-1}} : T_eG \to T_eG$ *such that*

$$\xi^r = Ad_g\, \xi^l\,. \tag{2.4}$$

The adjoint representation of G *is the set* $\{Ad_g : g \in G\}$.

The adjoint representation is a *representation* of the n-dimensional group G on the n-dimensional vector space T_eG; the theory of representations and their classification is important in quantum mechanics, but will not be used in the present work. Before presenting properties of Ad_g that are used for developments in the present work, additional structure must be introduced on the space T_eG.

The multiplication operator on G induces a *Lie bracket* operator on T_eG, which makes T_eG a *Lie algebra*.

Definition 2.4.7: *In general, a Lie algebra* $\mathfrak{g}$ *is a vector space associated with a* Lie bracket *operator* $[\,,\,] : \mathfrak{g} \times \mathfrak{g} \to \mathfrak{g} : \xi_1, \xi_2 \to \xi_3 = [\xi_1, \xi_2]$ *satisfying the following properties.*

- *The Lie bracket is linear in both of its arguments.*
- *The Lie bracket is anti-symmetric, i.e.* $[\xi_1,\, \xi_2] = -[\xi_2,\, \xi_1]\ \forall\ \xi_1, \xi_2 \in \mathfrak{g}$.
- *The Lie bracket satisfies the* Jacobi identity

$$[\xi_1,\, [\xi_2,\, \xi_3]] + [\xi_2,\, [\xi_3,\, \xi_1]] + [\xi_3,\, [\xi_1,\, \xi_2]] = 0 \quad \forall \xi_1, \xi_2, \xi_3 \in \mathfrak{g}\,.$$

Consider any trajectory $g(t)$ *on* G *with* $g(\tau) = e$ *and* $\frac{d}{dt} g|_\tau = \xi \ \in T_eG$. *Then with the Lie bracket defined by* $[\xi, \eta] = \frac{d}{dt}(Ad_g)|_\tau \eta$ *for any* $\eta \in T_eG$, *the tangent space* T_eG *to Lie group* G *at its identity* e *is the* Lie algebra $\mathfrak{g}$ associated to G.

The structure of Lie algebra $\mathfrak{g}$ associated to a Lie group G — i.e. the result of the Lie bracket of two elements in T_eG — is fully determined by the structure of G — i.e. the result of the multiplication of two group elements. Conversely, one Lie algebra can induce several different groups. In contrast to group multiplication, the Lie bracket is not invertible with respect to one of its arguments when the other is fixed. The Lie bracket characterizes the commutation properties of G: it can be seen as the "difference" between g_1g_2 and g_2g_1 for $g_1, g_2 \in G$ infinitesimally close to e.

Properties 2.4.8 (adjoint representation): *The adjoint representation Ad of a group G has the following properties.*

(a) $Ad_{g_1}Ad_{g_2} = Ad_{g_1g_2}$ *for any* $g_1, g_2 \in G$, $Ad_e =$ Identity *and* $Ad_{g^{-1}} = (Ad_g)^{-1}$ *for any* $g \in G$. *(This means that Ad is a* representation *of G.)*

(b) Ad_g *commutes with the Lie bracket, i.e.* $Ad_g\,[\xi_1, \xi_2] = [Ad_g\xi_1,\, Ad_g\xi_2]$ $\forall \xi_1, \xi_2 \in \mathfrak{g}$ *and* $\forall g \in G$.

(c) *Consider* $g(t)$ *with associated velocities* $\xi^l_g(t)$ *and* $\xi^r_g(t)$ *and denote by* $\xi^l_{g^{-1}}(t)$ *and* $\xi^r_{g^{-1}}(t)$ *the velocities associated to* $g(t)^{-1}$. *Then* $\forall g \in G$ *and* $\forall \xi_2 \in \mathfrak{g}$,

$$\xi^l_{g^{-1}}(t) = -Ad_{g(t)}\,\xi^l_g(t) \text{ and } \xi^r_{g^{-1}}(t) = -Ad_{g(t)^{-1}}\,\xi^r_g(t) \tag{2.5}$$

$$\tfrac{d}{dt}\left(Ad_{g(t)}\right)\xi_2 = Ad_{g(t)}\,[\xi^l_g(t),\, \xi_2] = [\xi^r_g(t),\, Ad_{g(t)}\,\xi_2] \tag{2.6}$$

$$\tfrac{d}{dt}\left(Ad^{-1}_{g(t)}\right)\xi_2 = -[\xi^l_g(t),\, Ad^{-1}_{g(t)}\,\xi_2] = Ad^{-1}_{g(t)}\,[\xi^r_g(t),\, \xi_2]\,. \tag{2.7}$$

Properties (2.6) and (2.7) imply $\frac{d}{dt}(\xi^l_g) = 0 \quad \Leftrightarrow \quad \frac{d}{dt}(\xi^r_g) = 0$.

Definition 2.4.9: *For a given velocity* $\xi \in \mathfrak{g}$, *the set* $O_\xi = \{Ad_g\,\xi : g \in G\}$ *is the* adjoint orbit *of* ξ.

The dimension of O_ξ may depend on ξ; in particular, $O_0 = \{0\}$ reduces to a point. Since Ad_g is invertible $\forall g \in G$, the adjoint orbits form a partition of $\mathfrak{g}$ into equivalence classes. Equation (2.4) implies that ξ^l and ξ^r always belong to the same adjoint orbit. The subspace of $\mathfrak{g}$ that is tangent to O_ξ at $\xi_1 \in O_\xi$ is $\xi_1 + T_{\xi_1}O = \xi_1 + \{[\xi_1, \zeta] : \zeta \in \mathfrak{g}\}$.

If G is an Abelian group, then
- $Ad_g =$ Identity $\forall g \in G \Rightarrow \xi^l = \xi^r$ and $O_\xi = \{\xi\}$;
- $[\xi_1, \xi_2] = 0$ $\forall \xi_1, \xi_2 \in \mathfrak{g}$.

2.4.2 Homogeneous manifolds

Homogeneous manifolds are a general class of highly symmetric manifolds, but that do not necessarily possess a Lie group structure allowing to combine two positions in order to build a third one, or to express all velocities in a particular tangent plane. The sphere for instance is a homogeneous manifold but not a Lie group. Intuitively, these are manifolds on which "all points are equivalent". It is thus relevant to study agreement (e.g. synchronization) problems for multiple agents evolving on such manifolds, since nothing allows a priori to prefer one point over another.

Definition 2.4.10: *A* homogeneous manifold $\mathcal{M}$ *is a manifold with a transitive group action by a Lie group G. It is isomorphic to the* quotient manifold *(see e.g. [2]) of G by one of its subgroups* G_1. *This quotient manifold* $\mathcal{M}$ *is canonically defined such that a point* $x \in \mathcal{M}$

corresponds to a set S_x of points of G, with g_1 and $g_2 \in G$ belonging to the same set S_x (thus the same point x on $\mathcal{M}$) if and only if they are equivalent modulo an action of G_1, i.e. if and only if $g_1^{-1} g_2 \in G_1$.

Any Lie group is a homogeneous manifold, since a Lie group is the quotient of itself by its trivial subgroup $\{e\}$ containing only the identity element. The converse however is not true, as exemplified by the two-dimensional sphere S^2 for instance. The sphere S^2 is the quotient of the Lie group $SO(3)$ of all three-dimensional rotations by the Lie group $SO(2)$ of rotations around a fixed axis, as explained in Section 2.4.4.

Homogeneous manifolds play an important role to solve some problems on Lie groups. For instance, for any Lie group G, the adjoint orbit O_ξ of any $\xi \in \mathfrak{g}$ is a homogeneous manifold. This is a direct consequence of Proposition 8.2.3.a in Part III of this dissertation.

2.4.3 Tangent vectors, distances and gradients on manifolds

As mentioned at the beginning of this section, the concepts are introduced here for the simpler case of manifolds *embedded in a vector space*, which is sufficient for the needs of the present dissertation. In particular, the distinction between vectors and co-vectors is not essential in this setting and is therefore not made.

Consider an n-dimensional smooth manifold $\mathcal{M}$ embedded in a vector space $\mathbb{R}^m$ with $m > n$. A proper definition of manifolds is beyond the scope of the present work. Locally, it is the result of a smooth application from $\mathbb{R}^n$ to $\mathbb{R}^m$; intuitive examples are a circle embedded in $\mathbb{R}^2$, a "curved" surface (maybe closed like a sphere) embedded in $\mathbb{R}^3$,... . Consider a smooth *curve* on $\mathcal{M}$ passing through $x_0 \in \mathcal{M}$, that is a smooth application $\gamma(t)$ from $(t_1, t_2) \subset \mathbb{R}$ to $\mathcal{M}$ and such that $\gamma(t_0) = x_0$ for some $t_0 \in (t_1, t_2)$.

Definition 2.4.11: *The* tangent vector *to γ at x_0 is the vector of $\mathbb{R}^m$ defined as the limit*

$$\tfrac{d}{dt}\gamma(t)|_{t=t_0} := \lim_{dt \to 0} \frac{\gamma(t_0 + dt) - \gamma(t_0)}{dt}$$

where $\gamma(t_a) - \gamma(t_b)$ is the usual difference of position vectors $\gamma(t_a)$ and $\gamma(t_b)$ in $\mathbb{R}^m$.

Definition 2.4.12: *The* tangent space $T_{x_0}\mathcal{M}$ *to manifold $\mathcal{M}$ at point x_0 is the space containing all possible tangent vectors at x_0 to smooth curves on $\mathcal{M}$ passing through x_0. It is an n-dimensional vector subspace of $\mathbb{R}^m$.*

The latter fact is interesting for computing tangent spaces to specific manifolds. For instance, on a unit-sphere S^2 embedded in $\mathbb{R}^3$ with its center at position 0, it suffices to consider two curves to deduce that the tangent space at $x \in S^2$ is the set of vectors y for which $x^T y = 0$.

Definition 2.4.13: *The* length *of a curve $\gamma : (t_1, t_2) \subset \mathbb{R} \to \mathcal{M}$ is defined as*

$$l(\gamma) = \int_{t_1}^{t_2} \| \tfrac{d}{dt}\gamma(t)|_\tau \| \, d\tau \ .$$

Definition 2.4.14: *The* geodesic distance *between two points a and b on $\mathcal{M}$ is defined as*

the minimum length of any curve between those points[1]*,*

$$d_{\mathcal{M}}(a,b) = \min_{\{\gamma:\gamma(t_a)=a \text{ and } \gamma(t_b)=b\}} \int_{t_a}^{t_b} \|\tfrac{d}{dt}\gamma(t)|_\tau\| \, d\tau \ .$$

The curve(s) γ *achieving this minimum is called the* shortest geodesic[2] *between* a *and* b*.*

It is important to note that the notion of distance between two points is not as obvious as it may seem at first sight. Nothing forbids to define the distance between two points a and b on $\mathcal{M}$ differently, for instance as the norm of the vector in $\mathbb{R}^m$ between a and b,

$$d_{\mathbb{R}^m}(a,b) := \|a-b\| \, .$$

The latter is used in Part II of the present dissertation, where it is called *chordal distance*. The geodesic distance is the canonical distance between points on $\mathcal{M}$. When $\mathcal{M} = S^1$ is the unit circle, the geodesic distance is the length of the shortest arc between a and b, while the chordal distance is the length of the chord joining a and b. If the geodesic distance is θ, the chordal distance is $2\sin(\theta/2)$.

Definition 2.4.15: *An* open convex set *on manifold* $\mathcal{M}$ *is an open set which contains all shortest geodesics between any of its points.*

For instance, a geodesic arc can be convex if it is not too long (e.g. shorter than π on the circle or sphere); a disconnected set is never convex; the largest possible open convex set on a sphere S^2 is the open hemisphere.

Consider a real function $f(x) : \mathcal{M} \to \mathbb{R}$. The *gradient* of f along $\mathcal{M}$ at x_0 is supposed to reflect how the value of f varies when its argument x moves away from x_0 along $\mathcal{M}$. Formally, the gradient of f at x_0 is a *cotangent vector* ζ^*, that is a linear application from $T_{x_0}\mathcal{M}$ to $\mathbb{R}$, such that $\frac{d}{dt}f(\gamma(t))|_{t=t_0} = \zeta^*(\xi)$ for any curve $\gamma(t)$ on $\mathcal{M}$ with $\gamma(t_0) = x_0$ and $\frac{d}{dt}\gamma(t)|_{t=t_0} = \xi \in T_{x_0}\mathcal{M}$. Using as a metric the canonical scalar product in $T_{x_0}\mathcal{M}$ induced by $\mathbb{R}^m$, as everywhere in the present dissertation, the gradient ζ^* can be identified with a vector $\zeta \in T_{x_0}\mathcal{M}$ for which $\zeta^*(\xi) = \zeta^T\xi$.

It is not difficult to show that, for $\mathcal{M} \equiv \mathbb{R}^m$, the gradient vector ζ corresponds to $\frac{\partial f}{\partial x}|_{x_0}$ as defined in Section 2.1; this justifies a posteriori the notation $\frac{\partial f}{\partial x} = \text{grad}(f)$, sometimes also written $\text{grad}_{\mathbb{R}^m}(f)$ to emphasize that the gradient is taken in $\mathbb{R}^m$. On a general manifold $\mathcal{M}$ embedded in $\mathbb{R}^m$, the possible motions of x away from x_0 are along curves on $\mathcal{M}$ through x_0; this means that, with respect to the gradient in the embedding space $\mathbb{R}^m$, the motion of x is just restricted to the tangent space $T_{x_0}\mathcal{M}$. This (simplified) argument leads to the following property of the gradient vector on manifolds, which will be used as a definition throughout the dissertation.

Definition 2.4.16: *The* gradient $\text{grad}_{\mathcal{M}}(f(x))|_{x_0}$ *along embedded manifold* $\mathcal{M} \subset \mathbb{R}^m$ *of a function* $f(x) : \mathcal{M} \to \mathbb{R}$*, evaluated at* x_0*, can be defined as the vector resulting from the orthogonal projection of* $\text{grad}_{\mathbb{R}^m}(f(x))|_{x_0}$ *onto the tangent space* $T_{x_0}\mathcal{M}$*.*

[1] If it is impossible to draw a smooth curve between a and b on $\mathcal{M}$, then $\mathcal{M}$ is said to be *disconnected*.

[2] Geodesics that are not "shortest" are the connection of shortest geodesics that have the same tangent vector at the connection point. For instance, on the sphere $S^2 \in \mathbb{R}^3$, any great circle arc is a geodesic, while only those arcs shorter than π are shortest geodesics.

Computing $\text{grad}_{\mathcal{M}}$ directly along the manifold can be much more efficient if the dimension of $\mathcal{M}$ is substantially lower than m. However, this requires more advanced tools of differential geometry. See [2] for an application of this technique to the Grassman manifold $Grass(p, n)$.

Definition 2.4.17: *A* gradient system *on a manifold is a dynamical system that can be written in the form* $\frac{d}{dt}x = \alpha \text{ grad}_{\mathcal{M}}(f(y))|_{y=x}$ *for some function* $f : \mathcal{M} \to \mathbb{R}$ *and constant* $\alpha \in \mathbb{R}$. *If* $\alpha > 0$ *the system is a* gradient ascent for f, *if* $\alpha < 0$ *it is a* gradient descent for f.

2.4.4 Some particular Lie groups and homogeneous manifolds

A recurrent vocabulary issue that may confuse the reader unfamiliar with the topic is the distinction between a Lie group or manifold and its *representation*. For the purpose of the present dissertation, a representation can just be seen as a convenient way to express the position on some manifold $\mathcal{M}$ by an explicit mathematical object, like a matrix or vector. It is sometimes preferred, especially for Lie groups, to represent an element of $\mathcal{M}$ by a matrix $\in \mathbb{R}^{n_1 \times n_2}$ instead of a vector $\in \mathbb{R}^m$. The developments for manifolds $\mathcal{M}$ embedded in $\mathbb{R}^m$ are adapted to this setting by assuming componentwise identification, for instance stacking the columns of a matrix $\in \mathbb{R}^{n_1 \times n_2}$ into a vector $\in \mathbb{R}^m$.

The vector space $\mathbb{R}^n$ can be viewed as a Lie group in the following way: group multiplication corresponds to addition of position vectors, the identity is the 0 vector and the group inverse of vector $x \in \mathbb{R}^n$ is vector $-x$. From the viewpoint of symmetry groups, $\mathbb{R}^n$ is associated to translation. It is an Abelian group.

Matrix Lie groups are Lie groups whose elements are represented as square matrices $\in \mathbb{R}^{n \times n}$ and for which group multiplication, identity and inverse correspond to the usual matrix multiplication, identity I_n and inverse. This allows to make computations as follows.

- The elements of Lie algebra $\mathfrak{g}$ are represented as matrices $\xi \in \mathbb{R}^{n \times n}$ (actually belonging to some subspace of $\mathbb{R}^{n \times n}$).
- Operations $L_{g*}\xi$ and $R_{g*}\xi$ are represented by matrix operations $g\xi$ and ξg respectively.
- The effects of the adjoint operator and Lie bracket are also simply deduced from matrix operations: $Ad_g\xi = g\xi g^{-1}$ and $[\xi_1, \xi_2] = \xi_1\xi_2 - \xi_2\xi_1$.

With some ingenuity, many Lie groups can easily be represented as matrix Lie groups (see e.g. $SO(n)$ and $SE(n)$ below).

The special orthogonal group $SO(n)$ is the group of rotations in $\mathbb{R}^n$. Equivalently, it can be viewed as the set of positively oriented orthonormal bases of $\mathbb{R}^n$; it is the natural state space for the orientation of a rigid body in $\mathbb{R}^n$. $SO(n)$ is a compact and connected Lie group. It has dimension $n(n-1)/2$.

In its canonical representation, used in the present dissertation, a point of $SO(n)$ is characterized by a real $n \times n$ orthogonal matrix Q with determinant equal to $+1$; thus $SO(n)$ is embedded in the vector space $\mathbb{R}^m = \mathbb{R}^{n \times n}$. Then $SO(n)$ is further characterized as follows.

- Group multiplication, inverse and identity are the corresponding matrix operations.
- The tangent space to $SO(n)$ at the identity I_n, i.e. its Lie algebra, is the space of antisymmetric $n \times n$ matrices $\mathfrak{so}(n)$, which also has dimension $n(n-1)/2$.

- Operations $L_{Q*}\xi$ and $R_{Q*}\xi$ are represented by $Q\xi$ and ξQ respectively, with $\xi \in \mathfrak{so}(n)$ and $Q \in SO(n)$. From this, the projection of $B \in \mathbb{R}^{n\times n}$ onto the tangent space $T_Q SO(n)$ to $SO(n)$ at Q is $Q(\frac{Q^T B}{2} - \frac{B^T Q}{2})$.
- The adjoint operator and Lie bracket are also simply defined with matrix products, $Ad_Q \xi_1 = Q\xi_1 Q^T$ and $[\xi_1,\, \xi_2] = \xi_1\xi_2 - \xi_2\xi_1$ with $\xi_1,\, \xi_2 \in \mathfrak{so}(n)$ and $Q \in SO(n)$.

The special Euclidean group $SE(n)$ is the group of all rigid body motions in $\mathbb{R}^n$, that is the set $\mathbb{R}^n \ltimes SO(n)$ of all n-dimensional rotation matrices and all n-dimensional translations; the product "$\ltimes$" is used instead of "$\times$" to emphasize that the rotation and translation components interact in the group structure. $SE(n)$ is not compact because $\mathbb{R}^n$ for the translation component is unbounded. It has dimension $n(n-1)/2 + n$.

In the present dissertation, a point of $SE(n)$ is written $g = (r, Q)$, where $r \in \mathbb{R}^n$ is the translation component and real orthogonal matrix $Q \in SO(n) \subset \mathbb{R}^{n\times n}$ is the rotation component. This can also be conveniently represented by a matrix in $\mathbb{R}^{(n+1)\times(n+1)}$ defined by

$$g = \begin{pmatrix} Q & r \\ \mathbf{0}_n & 1 \end{pmatrix} \quad ,$$

where r is considered as a column-vector and $\mathbf{0}_n$ is a row-vector containing n times 0. Then $SE(n)$ is seen as a matrix group, and matrix operations lead to its following characterization.

- Group multiplication: $g_1 g_2 = (r_1 + Q_1 r_2, Q_1 Q_2)$; identity: $e = (0, I_n)$; inverse: $g^{-1} = (-Q^T r, Q^T)$.
- The tangent space to $SE(n)$ at its identity e, i.e. its Lie algebra, contains elements $\xi = (v, [\omega]^\wedge) \in \mathfrak{se}(n)$, where $v \in \mathbb{R}^n$ and $[\omega]^\wedge \in \mathfrak{so}(n)$. The reason for notation $[\omega]^\wedge$ will become apparent when dealing with particular cases $SO(3)$ and $SE(3)$.
- Operations $L_{g*}\xi$ and $R_{g*}\xi$ yield $(Qv, Q[\omega]^\wedge)$ and $([\omega]^\wedge r + v,\ [\omega]^\wedge Q)$ respectively.
- The adjoint operator and Lie bracket yield $Ad_g\,(v, \omega) = (Qv - Q[\omega]^\wedge Q^T r,\, Q[\omega]^\wedge Q^T)$ and $[(v_1, \omega_1), (v_2, \omega_2)] = ([\omega_1]^\wedge v_2 - [\omega_2]^\wedge v_1, [\omega_1]^\wedge [\omega_2]^\wedge - [\omega_2]^\wedge [\omega_1]^\wedge)$.

Simplified expressions for $SE(2)$ and $SE(3)$ are provided in Part III of this dissertation.

The Grassmann manifold $Grass(p, n)$ is a manifold on which each point denotes a p-dimensional vector subspace $\mathcal{Y}$ of $\mathbb{R}^n$. The dimension of $Grass(p, n)$ is $p(n-p)$. Since $Grass(n-p, n)$ is isomorphic to $Grass(p, n)$ by identifying orthogonally complementary subspaces, this dissertation assumes without loss of generality that $p \leq \frac{n}{2}$. For the special case $p = 1$, the Grassmann manifold $Grass(1, n)$ is also known as the *projective space* in dimension n; a point on $Grass(1, n)$ represents a line in $\mathbb{R}^n$. $Grass(p, n)$ is connected and compact, but it is *in general* not a Lie group, although $Grass(1, 2) \cong SO(2)$ and $Grass(1, 4) \cong SO(3)$ for instance. However, every Grassmann manifold $Grass(p, n)$ is a homogeneous space, as the quotient of the orthogonal Lie group $O(n)$ by $O(p) \times O(n-p)$. Indeed, take a (not necessarily positively oriented) orthonormal basis $Q \in O(n)$ and say that $\mathcal{Y} \in Grass(p, n)$ is the subspace spanned by the first p column-vectors of Q. Then the same point $\mathcal{Y} \in Grass(p, n)$ is represented by any Q whose first p column-vectors span $\mathcal{Y}$ ($O(p)$-symmetry) and whose last $n-p$ column-vectors span the orthogonal complement of $\mathcal{Y}$ ($O(n-p)$-symmetry). Other quotient structures for $Grass(p, n)$ are discussed in [1].

A representation of $Grass(p, n)$ found in [1] assigns to $\mathcal{Y}$ any $n \times p$ matrix Y of p orthonormal column-vectors spanning $\mathcal{Y}$ (p-basis representation). All Y corresponding to rotations and reflections of the p column-vectors in $\mathcal{Y}$ represent the same $\mathcal{Y}$ ($O(p)$-symmetry), so this

representation is not unique. This means that $Grass(p,n)$ cannot be written as an embedded manifold with the p-basis representation. The dimension of this representation is $np - p(p+1)/2$. Alternatively, in [79], a point of $Grass(p,n)$ is represented by $\Pi = Y\,Y^T$, the orthonormal projector onto $\mathcal{Y}$ (projector representation). Using $\Pi_\perp = I_n - Y\,Y^T$, the orthonormal projector onto the space orthogonal to $\mathcal{Y}$, is strictly equivalent. The main advantage of the projector representation is that there exists a bijection between $Grass(p,n)$ and the orthonormal projectors of rank p, such that the projector representation makes $Grass(p,n)$ an embedded manifold of the cone $\mathbb{S}_n^+ \subset \mathbb{R}^{n\times n}$ of $n \times n$ symmetric positive semidefinite matrices. A disadvantage of this representation is its larger dimension $n(n+1)/2$.

It is shown in [79] that the projection of a matrix $M \in \mathbb{S}_n^+ \subset \mathbb{R}^{n\times n}$ onto the tangent space to $Grass(p,n)$ at a point $\mathcal{Y}$ represented by $\Pi_\mathcal{Y}$ is given by $\Pi_\mathcal{Y} M \Pi_{\perp\mathcal{Y}} + \Pi_{\perp\mathcal{Y}} M \Pi_\mathcal{Y}$.

The sphere S^n of dimension n is the set of points $x \in \mathbb{R}^{n+1}$ satisfying $\|x\| = 1$. This makes S^n an embedded manifold of $\mathbb{R}^{n+1}$. It is easy to see that the tangent space to S^n at $x \in S^n$ is composed of all vectors $y \in \mathbb{R}^{n+1}$ satisfying $x^T y = 0$. Then the projection of $b \in \mathbb{R}^{n+1}$ onto the tangent space to S^n at $x \in S^n$ is given by $b - x\,x^T b$.

S^n is compact and connected. It is generally not a Lie group (although $S^1 \cong SO(2)$), but well a homogeneous manifold. Indeed, consider a positively oriented orthonormal basis $Q \in SO(n+1)$ of $\mathbb{R}^{n+1}$. The first column-vector of Q can be said to represent a point $x \in S^n$, in which case all rotation matrices Q with the same first column-vector x must be considered equivalent. This amounts to quotienting $SO(n+1)$ by the Lie group $SO(n)$ of all possible rotations of the remaining basis vectors in the hyperplane orthogonal to x; thus $S^n \cong SO(n+1)/SO(n)$. Here, x and $-x$ as first column-vectors of $Q \in SO(n)$ represent different points, whereas for $Grass(1, n+1)$ they represent the same point (which follows from quotienting by $O(1) = \{1, -1\}$).

The circle is the simplest Lie group and nonlinear manifold. A point on the circle can be represented by an angle $\theta \in \mathbb{R}$ modulo 2π. Alternatively, the circle can be identified with the sphere S^1. A point is then represented by $x \in S^1 \subset \mathbb{R}^2$, with

$$x = \begin{pmatrix} \cos(\theta) \\ \sin(\theta) \end{pmatrix}.$$

The latter is the most useful representation of the circle, allowing an easy projection on the tangent space, as for the sphere. Sometimes, for the sake of shorter notation, the plane $\mathbb{R}^2$ is identified with the complex plane $\mathbb{C}$, with the first component corresponding to the real part and the second component corresponding to the imaginary part. Then $x = e^{i\theta}$.

The circle is also equivalent to the special orthogonal group $SO(2)$, since possible rotations in the plane describe a circle. With this representation, a point $x \in SO(2) \subset \mathbb{R}^{2\times 2}$ writes

$$x = \begin{pmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{pmatrix}$$

which depends 2π-periodically on the only parameter $\theta \in \mathbb{R}$ modulo 2π.

Finally, it is also possible to show that the circle is equivalent to $Grass(1,2)$. A point on $Grass(1,2)$ corresponds to a line in the plane. It is clear that the set of all lines in the plane can be represented by an angle $\phi \in \mathbb{R}$ modulo π, which contains the angle between the line in question and an arbitrary reference line. Thus choosing $\theta = 2\phi$, one obtains the manifold

represented by an angle $\theta \in \mathbb{R}$ modulo 2π, corresponding to the circle. In particular, the projector representation $\Pi_x \subset \mathbb{R}^{2\times 2}$ of $x \in Grass(1,2)$ then corresponds to

$$\begin{aligned}\Pi_x &= \begin{pmatrix} \cos(\phi) \\ \sin(\phi) \end{pmatrix} \begin{pmatrix} \cos(\phi) & \sin(\phi) \end{pmatrix} = \begin{pmatrix} \cos(\frac{\theta}{2}) \\ \sin(\frac{\theta}{2}) \end{pmatrix} \begin{pmatrix} \cos(\frac{\theta}{2}) & \sin(\frac{\theta}{2}) \end{pmatrix} \\ &= \begin{pmatrix} \cos^2(\frac{\theta}{2}) & \cos(\frac{\theta}{2})\sin(\frac{\theta}{2}) \\ \cos(\frac{\theta}{2})\sin(\frac{\theta}{2}) & \sin^2(\frac{\theta}{2}) \end{pmatrix} = \frac{I_2}{2} + \frac{1}{2} \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & -\cos(\theta) \end{pmatrix}\end{aligned}$$

which is again a 2π-periodic function of angle $\theta \in \mathbb{R}$ modulo 2π.

Part I

Synchronization on the circle

Main collaborators on this topic were Dr. L. Scardovi and Profs. V. Blondel, E. Tuna and N. Leonard. Related discussions repeatedly took place with Dr. J. Hendrickx.

Introduction The purpose of this first part is to present an in-depth study, with a geometric and constructive approach, of synchronization — i.e. driving all agents towards the same point — on the circle.

This problem setting contains most ingredients of synchronization problems on more general manifolds: a nonconvex but highly symmetric configuration space with global topology different from vector spaces, the limitation to relative positions in the controller, and the restricted communication among agents. Most issues that appear when trying to extend synchronization from vector spaces to manifolds can be illustrated on the circle. In fact, several results from Part I are generalized to homogeneous manifolds in Part II. The fact that the circle is just one-dimensional drastically reduces the complexity of technical details and therefore makes it an attractive candidate to introduce concepts. On the other hand, despite its apparent simplicity, synchronization on the circle already raises challenging questions — some of them still open — and several related results are difficult to extend to other manifolds. This justifies an in-depth geometric study of synchronization on the circle from a theoretical point of view.

As can be guessed from the vast literature on the subject, synchronization on the circle is also relevant in applications. Most prominently, the circle is the natural configuration space for the phase variables of oscillators, whose synchronization leads to collective rhythmic behavior; see for instance [138] for several examples ranging from laser technology to flashing fireflies. The famous Kuramoto model introduced in [71] to study such phenomena is directly related to the approach of this dissertation. There are other situations where the circle appears as a natural configuration space for synchronization, for instance in applications involving agreement on a direction in the plane. This directly connects to the problem of coordinated motion in the plane, which is studied in Part III of this dissertation. Vicsek's law is a popular model proposed in [148] to study synchronization in this framework, and is also directly related to the present approach. Physical positioning of agents on a circle is used for formations of underwater vehicles equipped with sensors to retrieve data in the ocean [76, 131, 132]. This particular application has led to a collaboration of Prof. R. Sepulchre at University of Liège with the team of Prof. N. Leonard at Princeton University, which was one of the main motivations for initiating the research work described in this dissertation. In fact, there is more to coordination on the circle than just synchronization: for instance, it may be required to distribute agents on the circle in some particular way. Such problems, although they already make an appearance in the present part, are typically related to the consensus issues discussed in Part II, where relevant applications are discussed. Other potential applications of synchronization and related problems on the circle can be found in the Introduction to the present work (Chapter 1).

Outline and Main points In line with the whole dissertation, Part I focuses on symmetries of the geometry: interaction among agents on the circle always involves *relative positions*, no reference position is introduced. To directly concentrate on geometric aspects, agent models are kept as simple as possible:

$$\tfrac{d}{dt}\theta_k = u_k \ \text{ (continuous-time)} \quad \text{or} \quad \theta_k(t+1) = \theta_k(t) + u_k \ \text{ (discrete-time)},$$

with u_k denoting the control input and θ_k the position on the circle.

Chapter 3 starts by introducing the synchronization problem with its main constraints (limited communication, equivalent agents, invariance with respect to absolute positions) on *vector spaces*. A corresponding linear consensus algorithm has been widely studied in the literature, with strong convergence results, and is a starting point for the extension to nonlinear manifolds in the present work. In a second section, the linear consensus algorithm is adapted to the circle, both in discrete- and continuous-time; this leads to "consensus algorithms" on the circle that have direct analogies with the Vicsek model [148] and the Kuramoto model [71] respectively. Convergence properties are examined, leading to positive conclusions as well as new difficulties with respect to vector spaces; the latter are illustrated by several examples.

Chapter 4 proposes several alternative algorithms to overcome the problems of the algorithms of Chapter 3; the discussion selects the most convenient setting between continuous- or discrete-time and sometimes assumes unweighted graphs, but corresponding results should be easy to obtain for other settings.

- A first alternative modifies the attraction law such that synchronization is the only stable equilibrium for any connected fixed undirected interconnection graph $\mathbb{G}$.
- A second alternative introduces randomness in the processing of information from neighboring agents and thereby achieves asymptotic synchronization with probability one for (possibly directed and varying) uniformly connected $\mathbb{G}$. A limitation of this algorithm is that its convergence rate can be extremely low and difficult to characterize. A specific directed version of the algorithm is completely independent of the underlying configuration space and can in fact be applied to reach agreement starting from any set of N initial symbols.
- A third alternative uses auxiliary variables to obtain an algorithm that rapidly converges to synchronization from almost all initial positions for uniformly connected $\mathbb{G}$.

Related literature The linear consensus algorithm on vector spaces has been proposed and studied by several authors, including [13, 94, 95, 104, 105, 143]; see [102] for a review. Regarding synchronization on nonlinear manifolds, the circle has attracted most of the attention. The most extensive literature on the subject derives from the Kuramoto model ([71], see [137] for a review), considering both position synchronization and velocity synchronization (here called coordinated motion, see Part III). An extensive amount of work has been — and is still being — published about synchronization phenomena of systems with periodic individual behavior, whose periodic orbit is topologically equivalent to the circle; see for instance [138] for a review of the diversified systems that are concerned. Recently, synchronization on the circle has been considered from a control perspective, the state variables representing headings of agents in the plane [148]. Most results cover *local* convergence [57, 95]. For all-to-all communication, (almost-)global convergence of the continuous-time consensus algorithm of Chapter 3 is proved in [131]. This result is extended to general interconnection graphs in [132] by using the algorithm of Section 4.3. The idea in Section 4.3 of using auxiliary variables to enhance synchronization under weak communication conditions was in fact first published in [118], and soon after that used by the authors of [130, 132] for coordinated motion of rigid bodies in the plane. The latter work directly relates to Part III of the present dissertation, in which synchronization on the circle can appear as a subproblem. Gossip algorithms, as used in Section 4.2, have been proposed and studied in vector spaces (see e.g. [17]).

Chapter 3

Synchronization: from vector spaces to the circle

Definition 3.0.18: *A swarm of N agents with states $x_k(t)$ evolving on a manifold, $k = 1, 2, ..., N$, is said to asymptotically* synchronize *or* reach synchronization *if*

$$x_1 = x_2 = ... = x_N \quad \textit{asymptotically.}$$

The asymptotic value of the x_k is called the consensus value.

3.1 Consensus on vector spaces

The material in this section was developed during the last decades; see [102] for a review. Applications where consensus on vector spaces can be used advantageously are also listed in [95] and other related work.

Consider a *swarm* of N agents with states $x_k \in \mathbb{R}^n$, $k \in \mathcal{V} = \{1, 2, ..., N\}$, evolving under continuous-time dynamics

$$\frac{d}{dt}x_k(t) = u_k(t) \ , \quad k = 1, 2..., N \tag{3.1}$$

or discrete-time dynamics

$$x_k(t+1) = x_k(t) + u_k(t) \ , \quad k = 1, 2..., N \tag{3.2}$$

where u_k is a control term. The goal is to design u_k such that the agents asymptotically synchronize, with the following restrictions on the controller.

1. Communication constraint: u_k may only depend on information concerning agent k and the agents $j \rightsquigarrow k$, which send information to k according to some imposed communication graph $\mathbb{G}$.
2. Configuration space symmetry: the behavior of the controlled swarm must be invariant with respect to uniform translation of all the agents: if $y_k(0) = x_k(0) + a \ \forall k \in \mathcal{V}$ for any $a \in \mathbb{R}^n$, then it must hold $y_k(t) = x_k(t) + a \ \forall k \in \mathcal{V}$ and $\forall t \geq 0$. Therefore u_k may only depend on *relative* positions of the agents, i.e. on $(x_j - x_k)$ for $j \rightsquigarrow k$. See Figure 3.2.(a).

3. Agent equivalence symmetry: all the agents in the swarm must be treated equivalently. This implies that (i) the form of u_k must be the same $\forall k \in \mathcal{V}$ and (ii) all $j \rightsquigarrow k$ must be treated equivalently in the control law of agent k.

This problem is traditionally called *consensus on a vector space*. On manifolds, the term "synchronization problem" is preferred because in Part II, the term "consensus" will be given a particular meaning different from synchronization. On vector spaces, "consensus" as defined in Part II is equivalent to synchronization, so both terms can be used interchangeably.

The consensus problem on vector spaces is solved by the linear controller

$$u_k(t) = \alpha \sum_{j=1}^{N} a_{jk}(t)(x_j(t) - x_k(t)) \ , \quad k = 1, 2..., N \tag{3.3}$$

where a_{jk} is the weight of link $j \rightsquigarrow k$ and α is a positive gain. Clearly, (3.3) satisfies the controller constraints. In continuous-time, the closed-loop system (3.1),(3.3) means that agent k is moving towards the position in $\mathbb{R}^n$ corresponding to the (positively weighted) arithmetic mean of its neighbors, $\frac{1}{d_k^{(i)}} \sum_{j \rightsquigarrow k} a_{jk} x_j$. In discrete-time, α must satisfy $\alpha\, d_k^{(i)}(t) \leq b$ for some constant $b < 1$, $\forall k \in \mathcal{V}$. Then (3.2),(3.3) means that the future position of agent k is at the (positively weighted) arithmetic mean $\frac{1}{\beta_k + d_k^{(i)}}(\sum_{j \rightsquigarrow k} a_{jk} x_j + \beta_k x_k)$ of its neighbors $j \rightsquigarrow k$ and itself, with non-vanishing weight β_k.

The convergence properties of the linear consensus algorithm on a vector space are well characterized. An extension of the following basic result in the presence of time delays can be found in [104]; the present dissertation does not consider time delays.

Proposition 3.1.1: *(adapted from [94, 95]) Consider a set of N agents evolving on $\mathbb{R}^n$ according to (continuous-time) (3.1),(3.3) with $\alpha > 0$ or according to (discrete-time) (3.2),(3.3) with $\alpha > 0$ and $\alpha d_k^{(i)}(t) \in [0, b]$ $\forall t \geq 0$, $\forall k \in \mathcal{V}$, for some constant $b \in (0, 1)$. Then the agents globally and exponentially converge to synchronization at some constant value $\bar{x} \in \mathbb{R}^n$ if and only if the communication among agents is characterized by a (piecewise continuous) δ-digraph which is uniformly connected.*

Proof: See [94, 95], or equivalently [13] or [143]. □

The proof of Proposition 3.1.1 essentially relies on the *convexity* of the update law, see Figure 3.1: the position of each agent k for $t > \tau$ always lies in the convex hull of the $x_j(\tau)$, $j = 1, 2, ..., N$. The permanent contraction of this convex hull, at a minimal rate because weights are non-vanishing, allows to conclude that the agents end up at a consensus value. An obvious negative consequence of Proposition 3.1.1 for non-varying $\mathbb{G}$ is that synchronization cannot be reached if $\mathbb{G}$ is not root-connected.

Proposition 3.1.2: *(see a.o. [104]) If in Proposition 3.1.1, the graph $\mathbb{G}$ is balanced for all times, then the consensus value is the arithmetic mean of initial values: $\bar{x} = \frac{1}{N} \sum_{k=1}^{N} x_k(0)$.*

Proof: It is easy to see that for a balanced graph, $\frac{1}{N} \sum_{k=1}^{N} x_k(t)$ is conserved over time. The conclusion is obtained by comparing its value for $t = 0$ and for t going to $+\infty$. □

If interconnections are not only balanced, but undirected and fixed, then the linear con-

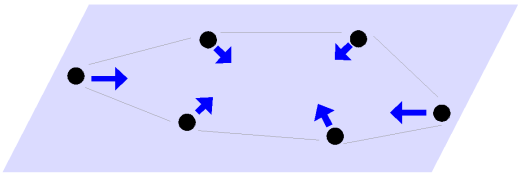


Figure 3.1: Consensus in the plane with algorithm (3.1),(3.3) or (3.2),(3.3): agents move towards a vertex or the interior of their convex hull; the latter therefore progressively shrinks in time, until reaching synchronization.

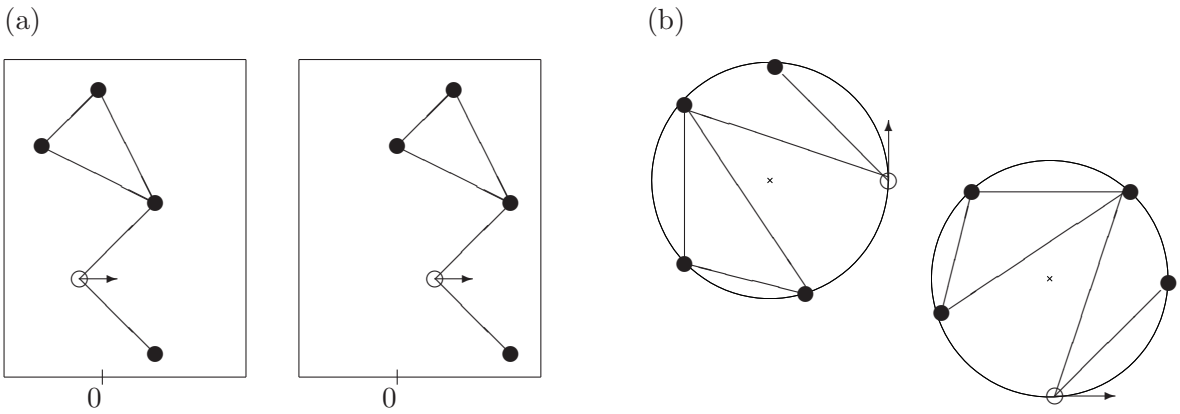


Figure 3.2: Invariance with respect to absolute position (a) on the plane and (b) on the circle. In both cases, the white agent's motion (the arrow) depends on the relative positions of its neighbors, such that its behavior remains identical when the whole swarm is moved uniformly.

sensus algorithm is a gradient descent algorithm for the cost function

$$V_{\text{vect}}(x) = \tfrac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\|x_j - x_k\|^2 = \|(B \otimes I_n)x\|^2 = x^T(L \otimes I_n)x \tag{3.4}$$

where B and L are the incidence and Laplacian matrices of $\mathbb{G}$ respectively and $x \in \mathbb{R}^{Nn}$ denotes the vector whose elements $(k-1)n+1$ to kn contain x_k.

3.2 Extension to the circle

Consider a swarm of N agents with states on the circle S^1. The global topology of the circle is fundamentally different from vector spaces, because if x_k denotes an angular position on the circle, then $x_k + 2\pi$ denotes the same position, whereas translating a point by a nonzero amount on a vector space necessarily brings it to a different position. This difference in topology, imposing a non-convex configuration space, is fundamental for the present work. To emphasize the difference with respect to vector spaces, the agents' states are now denoted θ_k, $k = 1, 2, ..., N$.

The agent dynamics remain the same, i.e. (3.1) or (3.2) with x_k replaced by θ_k. However, for the design of u_k, the different topology induces different implications of the configuration space symmetry. The behavior of the swarm must (i) be invariant with respect to a uniform translation of all θ_k and (ii) be invariant with respect to the translation of any single θ_k by a multiple of 2π — i.e., if $\phi_k(0) = \theta_k(0) + 2a\pi$ for some $k \in \mathcal{V}$ and $a \in \mathbb{Z}$, and $\phi_j(0) = \theta_j(0)$ $\forall j \neq k$, then it must hold $\phi_k(t) = \theta_k(t) + 2a\pi$ $\forall t \geq 0$ and $\phi_j(t) = \theta_j(t)$ $\forall j \neq k$ and $\forall t \geq 0$. This implies that u_k may only depend on *2π-periodic functions of the relative positions* $(\theta_j - \theta_k)$ of the agents $j \rightsquigarrow k$. See Figure 3.2.(b).

The simple linear algorithm (3.3) does not satisfy the periodicity required for configuration space symmetry, and therefore cannot be used on the circle. It can however be used to derive algorithms for synchronization on S^1 that are similar to (3.3) when all agents are within a small arc of the circle. The discrete-time and continuous-time cases are treated consecutively.

3.2.1 Discrete-time algorithm

Synchronization of $\theta_k \in S^1$, $k = 1, 2, ..., N$ can be seen as synchronization of $x_k \in \mathbb{R}^2 \cong \mathbb{C}$, $k = 1, 2, ..., N$ under the constraint $\|x_k\| = 1$. If the x_k were not restricted to $\|x_k\| = 1$, algorithm (3.2),(3.3) would synchronize them by imposing

$$x_k(t+1) = \frac{1}{\beta_k + d_k^{(i)}} \left(\sum_{j=1}^{N} a_{jk}\, x_j(t) \; + \beta_k\, x_k(t) \right) \;, \quad k = 1, 2, ..., N \;, \tag{3.5}$$

with some non-vanishing $\beta_k(t) > 0$. With this update law, $x_k(t+1)$ does generally not satisfy $\|x_k(t+1)\| = 1$. However, a corresponding point on the unit circle can be obtained by taking the argument of $x_k(t+1) \in \mathbb{C} \cong \mathbb{R}^2$. This leads to the following discrete-time algorithm for synchronization on the circle[1]

$$\theta_k(t+1) = \arg \left(\sum_{j=1}^{N} a_{jk}\, e^{i\theta_j(t)} \; + \beta\, e^{i\theta_k(t)} \right) \;, \quad k = 1, 2, ..., N \,, \tag{3.6}$$

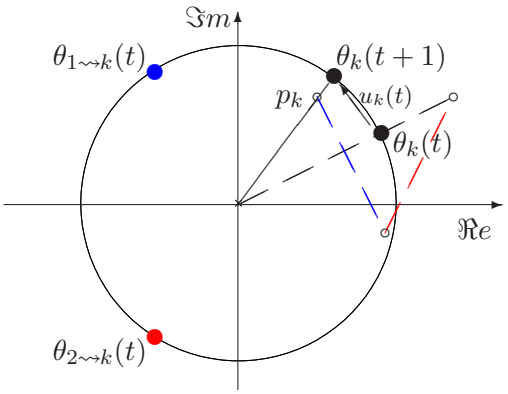
for some constant $\beta > 0$. The update of one agent according to this law is illustrated on Figure 3.3. It is clear from this picture that algorithm (3.6) respects the geometric invariance of configuration space S^1. This is confirmed mathematically by factoring out $e^{i\theta_k(t)}$ in (3.6) and rewriting it as

$$\theta_k(t+1) = \theta_k(t) + u_k = \theta_k(t) + \arg \left(\sum_{j=1}^{N} a_{jk}\, e^{i(\theta_j(t) - \theta_k(t))} \; + \beta \right) \;, \quad k = 1, 2, ..., N \,. \tag{3.7}$$

The feedback control term u_k indeed only depends on 2π-periodic functions of relative positions of connected agents $j \rightsquigarrow k$.

As for vector spaces, (3.6) is linked to a cost function when $\mathbb{G}$ is fixed undirected. Since $x_k(t+1)$ obtained with (3.5) is the endpoint of a gradient descent step for V_{vect}, then $\theta_k(t+1)$ is the projection on the unit circle of a point obtained by gradient descent in the complex plane, as a relaxation of $x_k = e^{i\theta_k}$.

[1] The present work takes the convention that arg(0) can take any value on S^1.



$$p_k := 1.5e^{i\theta_k} + e^{i\theta_1} + e^{i\theta_2}$$

Figure 3.3: Illustration of update law (3.6) for one agent k with $\beta = 1.5$ and $a_{1k} = a_{2k} = 1$.

The Vicsek model was first proposed in [148] to describe the discrete-time evolution of interacting particles that move with unit velocity in the plane. In the absence of noise, the update law of the Vicsek model is

$$r_k(t+1) = r_k(t) + e^{i\theta_k(t)} \tag{3.8}$$

$$\theta_k(t+1) = \arg\left(\sum_{j=1}^{N} e^{i\theta_j(t)} + e^{i\theta_k(t)}\right), \quad k = 1, 2, ..., N, \tag{3.9}$$

where θ_k denotes the heading angle of particle k and r_k its position in the plane. Heading update law (3.9) actually corresponds to (3.6) with $a_{jk} = \beta = 1$. The positions influence the dynamics of the headings through the interconnection graph $\mathbb{G}$: two agents j and k are interconnected at time t if and only if $\|r_k(t) - r_j(t)\| \leq R$ for some fixed $R > 0$; therefore $\mathbb{G}$ is called a *proximity graph*. The study of proximity graphs, or other state-dependent graphs, is beyond the scope of the present work.

Recently, [84] found the Vicsek law (3.8),(3.9) as one part of the primal/dual decomposition in an optimization algorithm for mobile antenna placement to achieve minimal communication power consumption.

3.2.2 Continuous-time algorithm

Taking the continuous-time limit of (3.6) amounts to letting the weight β, associated to the current position, grow indefinitely. In this limit case, $x_k(t+1) \in \mathbb{C}$ is defined in (3.5) with an infinitesimal gradient descent step for V_{vect}, and projected onto S^1 to yield $\theta_k(t+1)$. But this is strictly equivalent to projecting the gradient of V_{vect} onto the tangent to the circle at $\theta_k(t)$, and taking a corresponding infinitesimal descent step along the circle. The projection of $\frac{\partial V_{\text{vect}}}{\partial x_k}$ onto the tangent to the circle at x_k is, by definition, the *gradient of* V_{vect} *along the circle* associated to the embedding of S^1 in $\mathbb{C}$. Thus by viewing V_{vect} as a function of θ,

renamed for clarity

$$V_\theta(\theta) = \tfrac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\|e^{i\theta_j} - e^{i\theta_k}\|^2 = \tfrac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\left(2\sin(\tfrac{\theta_j-\theta_k}{2})\right)^2 , \tag{3.10}$$

the corresponding gradient descent algorithm along the circle, $\frac{d}{dt}\theta_k = -\alpha\frac{\partial V_\theta}{\partial \theta_k}\ \forall k \in \mathcal{V}$ with $\alpha > 0$, is the continuous-time limit of (3.6). Computing the gradient of (3.10) yields the following continuous-time algorithm for synchronization on the circle:

$$\tfrac{d}{dt}\theta_k = \alpha \sum_{j=1}^{N}(a_{jk}(t) + a_{kj}(t))\ \sin(\theta_j(t) - \theta_k(t)) , \quad k = 1, 2, ..., N \tag{3.11}$$

with constant $\alpha > 0$. This algorithm can only be implemented for undirected interconnection graphs: agent k would need to know the relative position of any agent $j \rightsquigarrow k$ *and* of any agent for which $k \rightsquigarrow j$; since information only comes from $j \rightsquigarrow k$, it is necessary that $j \rightsquigarrow k \Leftrightarrow k \rightsquigarrow j$. An extension of algorithm (3.11) to directed graphs is to write

$$\tfrac{d}{dt}\theta_k = 2\alpha \sum_{j=1}^{N} a_{jk}(t)\ \sin(\theta_j(t) - \theta_k(t)) , \quad k = 1, 2, ..., N . \tag{3.12}$$

This is the actual algorithm considered in the following. The factor 2 is for (3.12) to be the exact gradient of V_θ for undirected graphs. For directed graphs the gradient interpretation does not hold anymore. Algorithm (3.12) satisfies the geometric invariance of S^1, since the right-hand side is a 2π-periodic function of relative positions. With $x_k = e^{i\theta_k}$, (3.12) is equivalent to

$$\tfrac{d}{dt}x_k = 2\alpha\,\mathrm{Proj}_{x_k}\left(\sum_{j=1}^{N} a_{jk}\,(x_j - x_k)\right) \tag{3.13}$$

where $\mathrm{Proj}_{x_k}(r_k)$ denotes the orthogonal projection of $r_k \in \mathbb{C}$ onto the direction tangent to the unit circle at $x_k = e^{i\theta_k}$; identifying $\mathbb{C} \cong \mathbb{R}^2$ this means $\mathrm{Proj}_{x_k}(r_k) = r_k - x_k\,x_k^T r_k$. The geometric interpretation is that (3.13) defines a consensus update similar to (3.1),(3.3) but constrained to the subset of $\mathbb{R}^2$ where $\|x_k\| = 1$.

Algorithm (3.12) was already proposed in [129] which inspired the present work, and in connection with the Kuramoto model.

The Kuramoto model was first proposed in [71] to describe the continuous-time evolution of *phase variables*. It was inspired by the observation of spontaneous synchronization in several periodic natural phenomena, such as flashing fireflies or pendula suspended on the same support. Each agent k is considered as a periodic oscillator of natural frequency ω_k; without loss of generality, the *phase* of oscillator k at time t, i.e. the position on its cycle, can be denoted by variable $\theta_k \in S^1$. In addition to its natural evolution, each agent is attracted towards the position of all the others. The model proposed in [71] is

$$\tfrac{d}{dt}\theta_k = \omega_k + \sum_{k=1}^{N} \sin(\theta_j - \theta_k) , \quad k = 1, 2, ..., N . \tag{3.14}$$

Thus algorithm (3.12) derived above for synchronization on the circle in fact corresponds to the Kuramoto model with equal natural frequencies $\omega_1 = \omega_2 = ... = \omega_N$ (which, by change of variables, is equivalent to zero natural frequencies) but general interconnections among agents. This also highlights a link between the sine-model of Kuramoto and the "heading average" update law in the Vicsek model, which seems never to have been mentioned before.

When $\mathbb{G}$ is the complete graph, cost function V_θ can be rewritten

$$V_\theta = \tfrac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N}\|e^{i\theta_j} - e^{i\theta_k}\|^2 = N^2 - \left\|\sum_{k=1}^{N} e^{i\theta_k}\right\|^2 .$$

The quantity $\left\|\sum_{k=1}^{N} e^{i\theta_k}\right\|^2$ has been used for decades as a measure of the synchrony of phase variables in the literature on coupled oscillators. In the context of the Kuramoto model, it is known as the "complex order parameter".

The main point in studies of the Kuramoto model is the coordination of agents having different ω_k. The important issue of robustly coordinating agents despite their different natural tendencies is not the subject of the present dissertation.

3.3 Convergence properties

Section 3.2 proposes algorithms (3.6) and (3.12) as natural extensions of synchronization algorithms for the circle. However, the circle is not a convex configuration space, so the convergence properties of (3.6) and (3.12) are not as obvious as those of (3.1),(3.3) and (3.2),(3.3) on vector spaces. The purpose of the present section is to present positive convergence results, while Section 3.4 focuses on "negative" convergence results, i.e. situations in which asymptotic synchronization is not achieved.

3.3.1 Local synchronization like for vector spaces

When all agents are within a small subset of S^1, as on the left of Figure 3.4, then (3.12) becomes similar to (3.1),(3.3) because $\sin(\theta_j - \theta_k) \simeq (\theta_j - \theta_k)$ for small $(\theta_j - \theta_k)$. A similar observation can be made for the discrete-time algorithm. It is thus not surprising that [57, 95] are able to show that asymptotic synchronization on the circle is locally achieved under the same conditions as on vector spaces.

Proposition 3.3.1: *(adapted from [95]) Consider a set of N agents evolving on S^1 according to (continuous-time) (3.12) with $\alpha > 0$ or according to (discrete-time) (3.6) with $\beta > 0$. If the communication among agents is characterized by a (piecewise continuous) δ-digraph $\mathbb{G}$ which is uniformly connected and all agents are initially located within an open semicircle, then they exponentially converge to synchronization at some constant value $\bar{\theta} \in S^1$.*

Proof: (adapted from [95]) The proof is presented in continuous-time; it is similar in discrete-time, which is the original setting of [95]. To analyze the algorithm, assume without loss of generality that $\theta_k \in [-b, b] \subset (-\frac{\pi}{2}, \frac{\pi}{2})$ initially. Then $\sin(\theta_k(t) - \theta_j(t)) = c_{jk}(t)(\theta_j(t) - \theta_k(t))$ where $c_{jk}(t) \geq \frac{\sin(2b)}{2b} > 0$ depends on $(\theta_j(t) - \theta_k(t))$. Thus (3.12) is equivalent to

$$\tfrac{d}{dt}\theta_k = \alpha \sum_{j=1}^{N} a_{jk}(t)\, c_{jk}(t)\, (\theta_j(t) - \theta_k(t)) \tag{3.15}$$

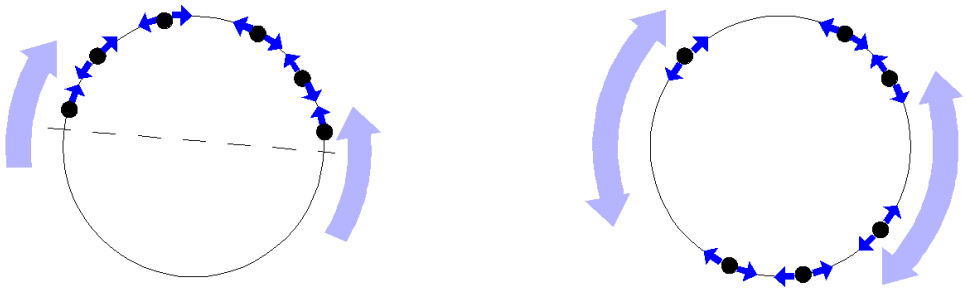


Figure 3.4: Motion of agents on the circle attracted towards the average position of their neighbors (synchronization algorithm (3.6) or (3.12)). Left: When all the agents are within a semicircle, convergence is ensured as on the real line. Right: When the agents are distributed over the whole circle, the collective behavior is a priori unclear and further analysis is required.

for some time-varying $c_{jk} \geq \frac{\sin(2b)}{2b} > 0$. Viewing the products $(a_{jk}c_{jk})$ as the weights of a δ_2-digraph $\mathbb{G}_2$, with $\delta_2 \geq \delta\frac{\sin(2b)}{2b}$, proves the Proposition. Indeed, (3.15) is strictly equivalent to (3.1),(3.3) with interconnection graph $\mathbb{G}_2$. The weights of $\mathbb{G}_2$ depend on the state of the agents, but its uniform connectivity is ensured if $\mathbb{G}$ is uniformly connected; then Proposition 3.1.1 can be applied. □

It is worth noting that in discrete-time, Proposition 3.3.1 requires no bound on β, in contrast to Proposition 3.1.1. For synchronization on $\mathbb{R}^n$ with (3.2),(3.3), the bound on α is necessary to ensure that the step taken by agent k "towards its in-neighbors" is small enough, to prevent k from actually going beyond its neighbors and so potentially leaving the convex hull of all x_j, $j = 1, 2, ..., N$. On the circle, β in (3.6) can be arbitrarily large[2]: then $\sum_{j=1}^{N} a_{jk}e^{i\theta_j} + \beta\, e^{i\theta_k}$ could be outside the convex hull of the $e^{i\theta_j}$, $j = 1, 2, ..., N$, but taking its argument ensures that $\theta_k(t+1)$ will be somewhere on the shortest arc containing all $\theta_j(t)$, $j = 1, 2, ..., N$ (when all agents are within a semicircle).

When the agents are distributed over more than a semicircle, as on the right of Figure 3.4, the proof of Proposition 3.3.1 no longer holds, because it is not ensured anymore that the c_{jk} will all be positive. This is in fact the consequence of a *loss of convexity*, implying that the strong vector space arguments of [94, 95] are no longer applicable. In algorithms (3.6) and (3.12), agents move towards their neighbors on the shortest path. Therefore, if the agents are initially located within an open convex set, then they remain within this set for all future times. An arc $s \subset S^1$ larger than a semicircle is not convex, because it does not contain the shortest path between points located close to the opposite extrema of the arc; a closed semicircle s is not convex because there are two equivalent shortest paths between its extreme points, namely s itself and the opposite semicircle. For agents distributed over more than a semicircle, it is thus easy to see that, a priori, they may leave any open arc $s \in S^1$ containing them all. However, further analysis allows to get the following positive convergence results.

[2]In the present setting, β is assumed constant in time; if β varies, it should stay bounded from below and from above, i.e. $\beta(t) \in [b_{min}, b_{max}]$ for some constant b_{min} and $b_{max} \in \mathbb{R}_{>0}$.

3.3.2 Convergence to local equilibria for fixed undirected $\mathbb{G}$

For fixed undirected $\mathbb{G}$, algorithms (3.6) and (3.12) can be seen as gradient descent systems for the cost function V_θ defined in (3.10). This allows to prove global convergence properties. The continuous-time algorithm has been analyzed in [131] for the complete graph.

Proposition 3.3.2: *Consider a set of N agents evolving on S^1 according to (3.12) with $\alpha > 0$, with communication graph $\mathbb{G}$ fixed and undirected. Then the agents always converge to a set of equilibria corresponding to the critical points of V_θ. The only asymptotically stable equilibria are the local minima of V_θ.*

Proof: The convergence of a continuous-time gradient system towards the critical set of its potential function V is a direct consequence of applying the LaSalle invariance principle with Lyapunov function V. For a descent algorithm, only local minima can be asymptotically stable: if there is a point $(\theta_1, \theta_2, ..., \theta_N)$ in the neighborhood of an equilibrium $(\bar{\theta}_1, \bar{\theta}_2, ..., \bar{\theta}_N)$ for which $V(\theta) < V(\bar{\theta})$, then from that point the descent algorithm can never come back to the equilibrium since this would require to increase V. □

The discrete-time algorithm has more particular convergence properties: in addition to a result similar to Proposition 3.3.2 but with a bound on β, it can be shown that under *locally asynchronous update*, convergence can be obtained for arbitrary β, i.e. without requiring any minimal inertia.

In discrete-time, agents are said to update *synchronously* if, between instants t and $t+1$, all agents $k \in \mathcal{V}$ apply the update algorithm (3.6). In contrast, agents are said to update *locally asynchronously* if, between instants t and $t+1$, only a subset of agents $\sigma \subset \mathcal{V}$ applies (3.6) and the others remain at their position, and set σ contains no agents that are connected to each other in $\mathbb{G}$. In an asynchronous setting, it must be ensured that every agent will eventually move over a uniform time horizon; this justifies the following assumption.

Assumption 3.3.3: *The sequence of index sets $\{\sigma(t)\}_{t_0 \leq t < \infty}$, $\sigma(t) \subset \mathcal{V}$ chosen for the locally asynchronous update of the agents has the property that there exist a finite time span T and a partition of the discrete-time space $[t_0, t_1)$, $[t_1, t_2)$,... with $(t_{n+1} - t_n) < T$ $\forall n \in \mathbb{Z}_{\geq 0}$, such that $k \in \cup_{t \in [t_n, t_{n+1})} \sigma(t)$ for every agent $k \in \mathcal{V}$ and for every $n \in \mathbb{Z}_{\geq 0}$.*

Proposition 3.3.4: *Consider a set of N agents evolving on S^1 by applying (3.6) locally asynchronously with $\beta > 0$, communication graph $\mathbb{G}$ fixed and undirected and update sequence satisfying Assumption 3.3.3. Then the agents almost always converge to a set of equilibria corresponding to the critical points of V_θ. The only asymptotically stable equilibria are the local minima of V_θ.*

Proof: The evolution $\Delta V_\theta(t)$ of V_θ between t and $t+1$ is the sum of the ΔV_θ implied by moving each agent in $\sigma(t)$ separately. Denote $\sum_{j=1}^{N} a_{jk} e^{i(\theta_j - \theta_k)} + \beta = \rho_k e^{i u_k}$. If only agent k

is updated at time t, then the variation of V_θ reduces to

$$\begin{aligned}\Delta V_\theta &= -2\sin(\tfrac{u_k}{2})\,\Im m\left(\sum_{j=1}^{N} a_{jk}\, e^{i(\theta_j-\theta_k-\frac{u_k}{2})}\right)\\ &= -2\sin(\tfrac{u_k}{2})\,\Im m\left(e^{-i\frac{u_k}{2}}\left(\sum_{j=1}^{N} a_{jk}e^{i(\theta_j-\theta_k)}+\beta\right)\right) - 2\beta\left(\sin(\tfrac{u_k}{2})\right)^2\\ &= -2(\rho_k+\beta)\,\left(\sin(\tfrac{u_k}{2})\right)^2 \le 0\,.\end{aligned}$$

This expression is also valid for $\rho_k = 0$. Thus V_θ can only decrease. Assumption 3.3.3 ensures that every agent k is updated at an infinite number of time instants $t_{1k}, t_{2k}, \ldots t_{nk}, \ldots$. As a consequence, since $V_\theta \ge 0$ and $\Delta V_\theta \le 0$, it is necessary that all sequences $\{\Delta V_\theta(t_{nk})\}$ converge to 0 when n goes to $+\infty$. This implies that either u_k or ρ_k must go to 0 $\forall k \in \mathcal{V}$. The case $\rho_k = 0$ has zero measure and can only appear "by chance", because it is the global maximum of V_θ with respect to θ_k. If u_k goes to 0, then agent k asymptotically approaches an equilibrium set. Like for Proposition 3.3.2, only minima can be asymptotically stable for a descent algorithm. □

For synchronous update, it is necessary to impose a bound on the motion of the agents, as on vector spaces. However, in the present setting on S^1, it is not so easy to find a minimal value for the inertia β to ensure that (3.6) remains a descent algorithm. The following result gives a conservative bound on β.

Proposition 3.3.5: *Consider a set of N agents evolving on S^1 by applying (3.6) synchronously with fixed, undirected and unweighted communication graph $\mathbb{G}$ and*

$$\beta \ge d_{max}\,(\tfrac{2}{M^*}+1) \quad \text{where} \quad \tfrac{e^{M^*}-1}{M^*} = 1+\tfrac{d_{max}}{d_{sum}}$$

with $d_{sum} = \sum_{j=1}^{N} d_j^{(i)}$ and $d_{max} = \max_{j\in\mathcal{V}}(d_j^{(i)})$. Then the agents almost always converge to a set of equilibria corresponding to the critical points of V_θ. The only asymptotically stable equilibria are the local minima of V_θ.

Proof: The proof shows that $\Delta V_\theta \le 0$ holds for synchronous operation and the bound on β. It consists of rather straightforward but lengthy computations and can be found in Appendix A.1. □

A problem with the bound of Proposition 3.3.5 is that each agent must know d_{sum} and d_{max}, which is information about the (communication structure of the) whole swarm.

The Hopfield model, proposed by Hopfield in [51] for a discrete-time network, also features the property of asynchronous convergence that fails to extend to synchronous operation. The Hopfield network considers N neurons with states $x_k \in \{-1, 1\}$. The discrete-time update law for the states of the neurons is

$$x_k(t+1) = \text{sign}\left(\sum_{j=1}^{N} a_{jk}x_j(t)+\xi_k\right)\,, \quad 1,2,\ldots,N \tag{3.16}$$

where ξ_k is a firing threshold for neuron k. Considering the cost function

$$V_H = \frac{-1}{2} \sum_{k=1}^{N} \sum_{j=1}^{N} a_{jk}\, x_j\, x_k - \sum_{k=1}^{N} x_k\, \xi_k \ ,$$

[51] shows that when (3.16) is applied asynchronously with a random update sequence, the property $V_H(t+1) \leq V_H(t)$ always holds and the network eventually reaches a fixed point that corresponds to a local minimum of V_H. This is not true anymore for synchronous operation. In fact, it is shown in [42] that the system can go into a limit cycle in that case.

This compares well with algorithm (3.6): both laws are the projection of descent algorithms for a symmetric quadratic potential to states restricted to a subset of an Euclidean state space[3]. In both situations, convergence is achieved by asynchronously moving agent k towards the point in its state space which is closest to some point p_k in the associated Euclidean space ($\mathbb{C}$ or $\mathbb{R}$); this point p_k is defined as the weighted mean of the neighbors of agent k. The absence of inertia in (3.16) would correspond to $\beta = 0$ in (3.6); then both algorithms fail to converge in synchronous operation. Section 3.4 shows that, like (3.16), in the absence of inertia, (3.6) can lead to a limit cycle in synchronous operation (at least for some $\mathbb{G}$).

Propositions 3.3.2, 3.3.4 and 3.3.5 say that the stable equilibria are the minima of V_θ. Unfortunately, as will be shown in Section 3.4, V_θ can have local minima different from synchronization for fixed connected undirected graphs $\mathbb{G}$. Then synchronization is only locally asymptotically stable. For some graphs however, synchronization is (almost) globally asymptotically stable.

3.3.3 Some graphs ensure (almost) global synchronization

Proposition 3.3.6: *(adapted from [24, 131]) If $\mathbb{G}$ is an undirected tree, the complete graph, or any vertex-interconnection of trees and complete graphs[4], then synchronization is the only local minimum of V_θ.*

Proof: The proof that synchronization is the only minimum of V_θ for vertex-interconnections of graphs for which the only minimum of V_θ is synchronization can be found in [24]. For the particular cases of the complete graph and the tree, the proof is relatively simple.

For the complete graph, V_θ can be written

$$V_\theta = N^2 - (p_\theta)^2 = N \sum_{k=1}^{N} \|x_k - \tfrac{p_\theta}{N}\|^2 \quad \text{where} \quad p_\theta := \sum_{j=1}^{N} e^{i\theta_j} \ .$$

If $p_\theta = 0$, then the state is a local maximum of V_θ, thus unstable. If $p_\theta \neq 0$, then $\forall k \in \mathcal{V}$, a critical point of V_θ requires either $\theta_k = \arg(p_\theta)$ or $\theta_k = \arg(p_\theta) + \pi$. In the second case, any motion of agent k decreases V_θ, so it is unstable. In conclusion, for stable equilibria, $\theta_1 = \theta_2 = ...\theta_N = \arg(p_\theta)$.

[3] The set $\{-1, 1\}$ can be considered as the "sphere of dimension 0" by defining the sphere of dimension $n-1$ as the set of points in $\mathbb{R}^n$ that are at Euclidean distance 1 from the origin. The circle is the sphere of dimension 1.

[4] A vertex-interconnection of two graphs $\mathbb{G}_1(\mathcal{V}_1, \mathcal{E}_1)$ and $\mathbb{G}_2(\mathcal{V}_2, \mathcal{E}_2)$ is a graph $\mathbb{G}$ whose vertices can be partitioned into a singleton $\{k\}$ and two sets $\mathcal{V}_a$, $\mathcal{V}_b$ and whose edge set can be partitioned into two sets $\mathcal{E}_a$, $\mathcal{E}_b$, such that $\mathcal{V}_a \cup \{k\} = \mathcal{V}_1$, $\mathcal{E}_a = \mathcal{E}_1$, $\mathcal{V}_b \cup \{k\} = \mathcal{V}_2$ and $\mathcal{E}_b = \mathcal{E}_2$.

For a tree, start by considering motions of a leaf l. V_θ characterizes the distance towards its parent p, which is maximized for $\theta_l = \theta_p + \pi$ and minimized for $\theta_l = \theta_p$; thus only the case where all leaves are synchronized with their parents must be retained as a potential minimum. Now consider an agent p whose children are all leaves; from the previous assertion, it need only be considered in the situation synchronized with all its children. But considering synchronous motion of p and its children, V_θ characterizes the distance from p to its parent, which yields the same case as with the leaves. This argument can be repeated until the whole tree has been covered, implying that all agents must be synchronized at a minimum of V_θ. □

Although Propositions 3.3.2, 3.3.4 and 3.3.5, ensuring convergence to a local minimum of V_θ, require $\mathbb{G}$ to be undirected, specific graphs ensuring almost global synchronization can also be directed.

Proposition 3.3.7: *Consider a set of N agents evolving on S^1 by applying (3.12) with $\alpha > 0$ or (3.6) with $\beta > 0$. If communication graph $\mathbb{G}$ is a fixed, directed root-connected tree, then the agents asymptotically converge to synchronization at the initial position of the root, for almost all initial conditions.*

Proof: Under (3.12) or (3.6), a child c of the root r will move towards the root and reach it asymptotically, unless $\theta_c = \theta_r + \pi$. The childs of c behave similarly with respect to c; the argument can be pursued down to the leaves. □

3.4 Obstacles to global synchronization

Section 3.3.2 identifies situations where the synchronization algorithms on S^1 converge to a set of equilibria. But this does not imply that convergence is necessarily towards synchronization. Section 3.4.1 looks more carefully at equilibria of (3.6) and (3.12) different from synchronization. In other situations, (3.6) and (3.12) do not even converge to an equilibrium set. Several examples illustrating this fact are provided in Section 3.4.2. Finally, a few words are said about the case of *state-dependent interconnection graphs*, which are not further addressed in the present dissertation.

3.4.1 Stable equilibria different from synchronization for fixed graphs

Definition 3.4.1: *A* configuration *is a particular set of relative positions of the agents. Thus a configuration is equivalent to a point $(\bar\theta_1, \bar\theta_2, ..., \bar\theta_N) \in S^1 \times S^1 \times ... \times S^1$ and all the points obtained by its uniform translations $(\bar\theta_1 + a, \bar\theta_2 + a, ..., \bar\theta_N + a)$ for $a \in S^1$.*

First consider the case of undirected graphs. Propositions 3.3.2, 3.3.4 and 3.3.5 ensure that the agents asympototically converge to the set of critical points of V_θ, and only local minima of V_θ are stable. Depending on $\mathbb{G}$, there may be local minima different from synchronization. The following example can also be found in [59].

Ex. 3.4.2: equilibria for the undirected ring: Consider a set of N agents interconnected according to an undirected, unweighted ring graph. Then the critical points of V_θ must satisfy $\sin(\theta_{j_a(k)} - \theta_k) + \sin(\theta_{j_b(k)} - \theta_k) = 0$ where $j_a(k)$ and $j_b(k)$ are the two neighbors of k in the ring graph. This requires that the positions of consecutive agents in the ring graph differ

either by θ_0 or by $\pi - \theta_0$, for some $\theta_0 \in [0, \frac{\pi}{2}]$ well chosen such that the sum of all angle differences is a multiple of 2π.

Stability of these equilibria can be assessed by examining the Hessian of V_θ. At any point of $S^1 \times S^1 \times ... \times S^1$, the Hessian has one zero eigenvalue with eigenvector $\mathbf{1}_N$, corresponding to a uniform translation of all agents ($\theta_k \to \theta_k + a \ \forall k \in \mathcal{V}$, for some $a \in S^1$). If all other eigenvalues of V_θ are positive, then the configuration is a minimum of V_θ; if they are all negative, it is a maximum, and if there are negative and positive eigenvalues, it is a saddle point. The minimum and maximum eigenvalues of a symmetric matrix H are the extreme values of the *Rayleigh quotient* $r(x) = \frac{x^T H x}{x^T x}$ for $x \in \mathbb{R}^N$. The Hessian H_θ of V_θ has elements

$$(h_\theta)_{jk} = \begin{cases} \sum_{j=1}^N a_{jk} \cos(\theta_j - \theta_k) & \text{for } j = k \\ -\cos(\theta_j - \theta_k) & \text{if } a_{jk} = 1 \\ 0 & \text{if } a_{jk} = 0\,. \end{cases}$$

The particular value θ_0 or $\pi - \theta_0$ of $|\theta_j - \theta_k|$ at critical points of V_θ leads to the following.

- If $|\theta_j - \theta_k| = \theta_0 \neq \frac{\pi}{2}$ for some $(j,k) \in \mathcal{E}$ and $|\theta_j - \theta_k| = \pi - \theta_0$ for other $(j,k) \in \mathcal{E}$, then the off-diagonal elements of H_θ take values 0, $+\cos(\theta_0)$ and $-\cos(\theta_0)$. Then it is easy to build vectors $x_1 \in \mathbb{R}^N$ and $x_2 \in \mathbb{R}^N$ such that $x_1^T H_\theta x_1 > 0$ and $x_2^T H(\theta) x_2 < 0$ (see [59]), so this situation is a saddle.
- If $|\theta_j - \theta_k| = \theta_0 \in [0, \pi] \setminus \{\frac{\pi}{2}\}\ \forall (j,k) \in \mathcal{E}$, then the Hessian equals $\cos(\theta_0)$ times the Laplacian of an undirected ring graph. Since the latter is positive semidefinite, the equilibrium is a minimum of V_θ if $\theta_0 < \frac{\pi}{2}$ and a maximum of V_θ if $\theta_0 > \frac{\pi}{2}$.
- If $\theta_0 = \frac{\pi}{2}$, then the Hessian is zero. However, leaving one agent fixed, moving one of its neighbors by $\delta\theta$ and all other agents by $-\delta\theta$, the cost V_θ decreases, so this cannot be a local minimum of V_θ.

In summary, the minima of V_θ are configurations with agents distributed on the circle such that consecutive agents in the ring graph are separated by θ_0, for some $\theta_0 \in (-\frac{\pi}{2}, \frac{\pi}{2})$ satisfying $N\theta_0 = 2a\pi$ with $a \in \mathbb{Z}$. The case $\theta_0 = 0$ corresponds to synchronization. When $N \geq 5$, it is possible to find stable configurations with $\theta_0 \neq 0$, see Figure 3.5. For all these configurations, $\sum_{k=1}^N e^{i\theta_k} = 0$, therefore they are said to be *balanced*. The particular configurations where the distance between any pair of consecutive agents on S^1 is $\frac{2\pi}{N}$ (see Figure 3.5) are called *splay states* in [131]. ◇

As a conclusion, for some graphs (e.g. complete graph, tree), synchronization is the only minimum of V_θ, while others (e.g. undirected ring) admit additional local minima. The importance of local minima of V_θ as spurious stable equilibria of (3.6) or (3.12) motivates the following definition.

Definition 3.4.3: *A fixed undirected graph $\mathbb{G}$ is called* S^1-synchronizing *if it admits no local minima of V_θ different from synchronization.*

O.Q.: The question of characterizing[5] which graphs are S^1-synchronizing is currently open.

Figure 3.6 illustrates, classifying some small undirected unweighted graphs, that similar graphs can have different S^1-synchronizing properties. The undirected ring graphs can be transformed into a tree, which is S^1-synchronizing, by deleting a single edge.

[5] with simple graph properties like e.g. the presence of certain cliques, cycles or subgraphs, in contrast to

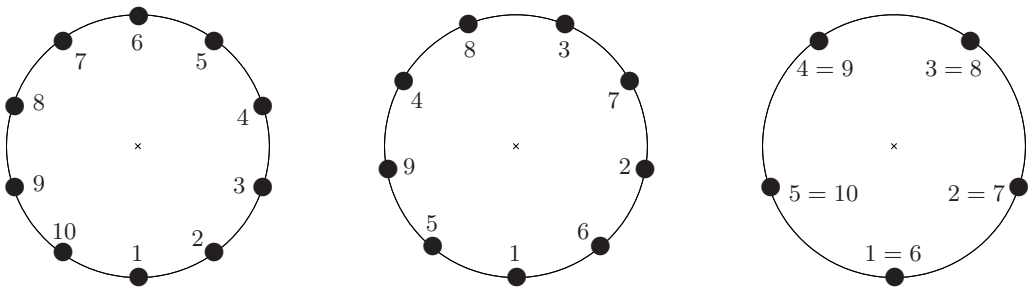


Figure 3.5: Several balanced configurations that are minima of V_θ for the undirected ring graph, with 10, 9 and again 10 agents. Agents are numbered in the order of the ring, e.g. agent 3 is connected to agents 2 and 4. The two left plots display splay states.

The following result shows that stable configurations different from synchronization are not exceptional: in fact, any configuration that is sufficiently "spread" on the circle is stable under the synchronization algorithms for a well-chosen weighted digraph.

Proposition 3.4.4: *Consider a set of N agents distributed on S^1 in a configuration $\{\theta_k\}$ such that for every k, there is at least one agent located in $(\theta_k, \theta_k + \pi/2)$ and one agent located in $(\theta_k - \pi/2, \theta_k)$; such a configuration requires $N \geq 5$. Then there exists a positively weighted and strongly connected δ-digraph making this configuration locally exponentially stable under (3.12) with $\alpha > 0$.*

Proof: From (3.13), a necessary and sufficient condition for a configuration to be an equilibrium is that $r_k := \sum_{j=1}^N a_{jk}\,(x_j - x_k)$ must be aligned with $x_k = e^{i\theta_k}$. Since the positive linear combinations of two vectors in the plane span the cone between them, one can assign positive weights to edges (j,k) with $\theta_j \in (\theta_k - \pi/2, \theta_k + \pi/2)$, and zero weight to all other edges (m,k) towards k, such that r_k and x_k are aligned, for each k. This ensures that the configuration is an equilibrium. Obviously, the graph $\mathbb{G}$ associated to positive weights is strongly connected. Linearization of (3.12) around the equilibrium yields a linear consensus algorithm in $\mathbb{R}$ with weights $a_{jk}\cos(\theta_j - \theta_k) \in \{0\} \cup [\gamma_-, \gamma_+]$, where $\gamma_- = \min_{(j,k)\in\mathcal{E}(\mathbb{G})}(a_{jk}\cos(\theta_j - \theta_k)) > 0$ and $\gamma_+ = \max_{(j,k)\in\mathcal{E}(\mathbb{G})}(a_{jk}\cos(\theta_j - \theta_k)) < +\infty$. According to Proposition 3.1.1, this ensures exponential stability of the equilibrium configuration (modulo a uniform translation of all agents). □

Proposition 3.4.4 identifies how specific configurations can be made locally exponentially stable by choosing appropriate weights for directed graph edges. For any of these choices, synchronization is also exponentially stable — but thus only locally. It can sometimes be interesting to stabilize configurations different from synchonization, like splay states for instance (see Figure 3.5). Variants of algorithm (3.12) that *prevent* a swarm of agents on S^1 from converging to synchronization, and rather stabilize alternative *balanced* configurations (see Example 3.4.2), are proposed in [131, 132]. These alternatives consist of specific poly-

the definition which requires to compute all local minima of V_θ

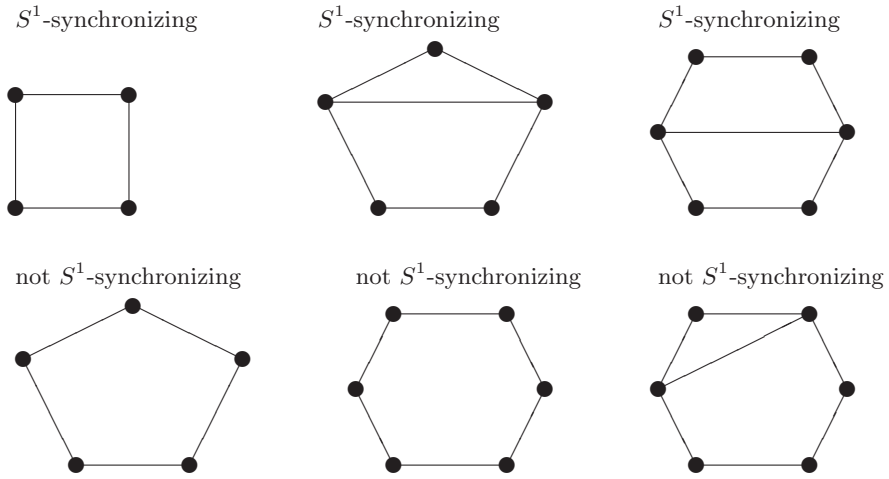


Figure 3.6: A few undirected unweighted graphs that are S^1-synchronizing or not.

nomials in $\sin(\theta_j - \theta_k)$, with in particular a negative gain on the first-order term, i.e. $\alpha < 0$ instead of $\alpha > 0$ like in (3.12). Part II of the present dissertation discusses in more detail such configurations different from synchronization.

3.4.2 Limit sets different from equilibrium

Section 3.4.1 lists cases where (3.12) or (3.6) do not converge to *synchronization*, but still to a set of equilibria. There are also cases where the agents do not converge to a set of equilibria.

A first remark concerns the behavior of discrete-time algorithm (3.6) in synchronous operation (all agents update at the same time) for fixed undirected $\mathbb{G}$, but with arbitrarily small β; the bound of Proposition 3.3.5 is not tight, and the following illustration considers $\beta = 0$.

Consider the opposite individual behavior to (3.6), namely "moving away from" neigboring agents; the corresponding algorithm is

$$\theta_k(t+1) = \arg\left(\beta e^{i\theta_k(t)} - \sum_{j=1}^{N} a_{jk}\, e^{i\theta_j(t)}\right) \;, \quad k = 1, 2, ..., N \;, \tag{3.17}$$

which would be a gradient ascent for V_θ when β is sufficiently large. But when $\beta = 0$, algorithm (3.6) and its "opposite" algorithm (3.17) are strictly equivalent regarding evolution of relative positions. Indeed, since $\arg(-a) = \arg(a) + \pi$ for $a \in \mathbb{C}$, the only difference between (3.6) and (3.17) is that in (3.17), all agents make an additional turn by π. This changes nothing to the *relative* positions of the agents, so the evolution of all relative positions is the same for (3.6) as for (3.17). In simulation, one observes that the agents' behavior depends on $\mathbb{G}$ and initial conditions. For some $\mathbb{G}$ and initial conditions, under (3.6) the agents converge to a minimum of V_θ with fixed positions on S^1, while under (3.17) they reach the same configuration but with agents turning by π at each time step. For other initial conditions, under (3.17) the agents converge to a local maximum of V_θ, while (3.6) drives them to the same configuration but with agents turning by π at each time step. Taking $\beta = 0$ in (3.6) can also

lead to stable limit cycle behavior, as illustrated on Figure 3.8 (stability can be proved by similarity with Example 3.4.2).

Interconnection graphs that are not fixed and/or not undirected have even less clear convergence properties, since the corresponding algorithms do not retain the gradient property.

Periodic and quasi-periodic behaviors can easily be constructed for (3.12) with fixed *directed* graphs. The simplest such behavior is called *cyclic pursuit*: each agent k is attracted by its neighbors to move (say) clockwise, and the agents keep turning without synchronizing. A classical situation of stable cyclic pursuit is a directed ring graph with consecutive agents separated by $\frac{2\pi}{N} < \frac{\pi}{2}$. For agents moving in cyclic pursuit, the relative positions remain constant and the angular velocity stabilizes at a fixed value. However, examples of more exotic behavior can be built.

Ex. 3.4.5: periodic and non-periodic variation of relative positions for fixed directed $\mathbb{G}$ A periodic behavior of the configuration occurs when *relative positions periodically vary* in time. Such a situation can be built with agents partitioned into two sets such that each set is in cyclic pursuit at a different velocity. Start for instance with two unweighted directed ring graphs of N_1 and N_2 agents, where $N_1 + N_2 = N$ and $N_1 > N_2$; the positions of consecutive agents in each ring are separated by $\frac{2\pi}{N_1}$ and $\frac{2\pi}{N_2} < \frac{\pi}{2}$. Then the resulting behavior is that the second ring moves $\frac{N_1}{N_2}$ times faster than the first ring, so the relative position between agents belonging to different rings varies. Another situation where relative positions vary periodically is when the two sets move in opposite directions, as depicted on Figure 3.7. In both cases, with the edges proposed so far, the overall graph is not connected. To obtain a strongly connected graph, each agent of the first ring can be coupled to all the agents of the second ring and conversely; indeed, for a set of regularly spaced agents $\sum_k e^{i\theta_k} = 0$, so coupling an agent to all agents in such a set does not change its behavior.

Likewise, a *quasi-periodic variation* of relative positions is obtained when several sets of agents move in cyclic pursuit with irrational velocity ratios. This can be built with unitary graph weights and $\alpha = 1$. For instance, a first set has x agents in a splay state for an undirected ring graph $\Leftrightarrow \frac{d}{dt}\theta_k = 0$, a second set has 6 agents in the classical situation of cyclic pursuit with a directed ring graph $\Leftrightarrow \frac{d}{dt}\theta_k = \sqrt{3}$, and a third set has 12 agents in the classical situation of cyclic pursuit with a directed ring graph $\Leftrightarrow \frac{d}{dt}\theta_k = 1$.

The motion of the agents in the previous situations is still very regular. An example of *disorderly-looking quasi-periodic motion* can be built by adding to the last situation an agent which has no out-neighbors but is influenced by one agent in each of the three rings; the motion of this agent is illustrated on Figure 3.9. ◇

Ex. 3.4.6: reversal of direction of motion under fixed directed $\mathbb{G}$ It is possible to build situations of fixed directed coupling with even more surprising behavior. Figure 3.10 represents the motion of two sets of agents in cyclic pursuit, with coupling among agents of the two sets and initial positions such that all the agents periodically revert their direction of motion. See the caption of Figure 3.10 for more details on the setup. ◇

For time-varying graphs, the situation is even more complicated. In contrast to the vector

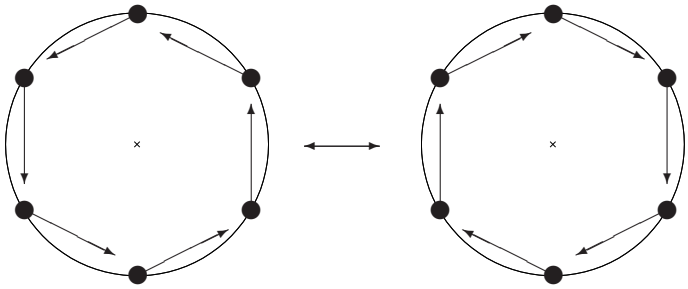


Figure 3.7: Periodic variation of relative positions on the circle for a fixed directed graph. A first set of agents turns clockwise in cyclic pursuit, a second set turns counterclockwise in cyclic pursuit; the two sets are represented on separate circles for better visibility. To obtain a strongly connected graph, in addition all the agents of the first set are connected with an undirected edge to all the agents of the second set.

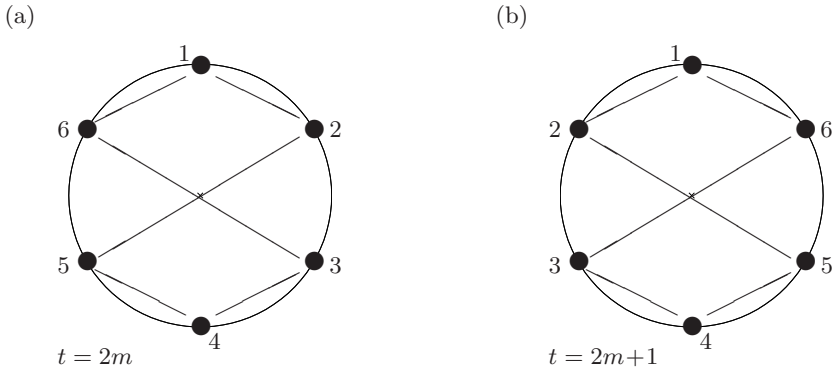


Figure 3.8: Illustration of limit cycle behavior of (3.6) with $\beta = 0$ and fixed undirected $\mathbb{G}$: (a) agent positions at even time instants. (b) agent positions at odd time instants.

space algorithms, behavior of the circle algorithms (3.6) and (3.12) strongly depends on the sequence of interconnection graphs $\mathbb{G}$. Indeed, since many different configurations can be stable on the circle depending on $\mathbb{G}$, the swarm can be driven towards different equilibria during longer or shorter time spans, implying no particular characterization of the swarm's behavior if $\mathbb{G}(t)$ can be arbitrary. In practice, synchronization is often eventually observed. This is because synchronization is ensured for connected graph sequences as soon as agents all lie in the same semicircle, see Proposition 3.3.1. But other asymptotic behaviors are possible. Just as a particular illustration, the following shows how limit cycle-like behavior of the relative positions can occur with undirected time-varying $\mathbb{G}$.

Ex. 3.4.7: limit cycle for undirected varying $\mathbb{G}$ Consider a set of agents interconnected according to an undirected ring $\mathbb{G}_1$ as in Figure 3.11.(a) and close to the local minimum of V_θ with consecutive agents separated by $\phi = \frac{2\pi}{N}$ (splay state). Applying (3.6) or (3.12) with $\mathbb{G}_1$ drives the swarm closer to this splay configuration (Figure 3.11.(b)). Now at some time,

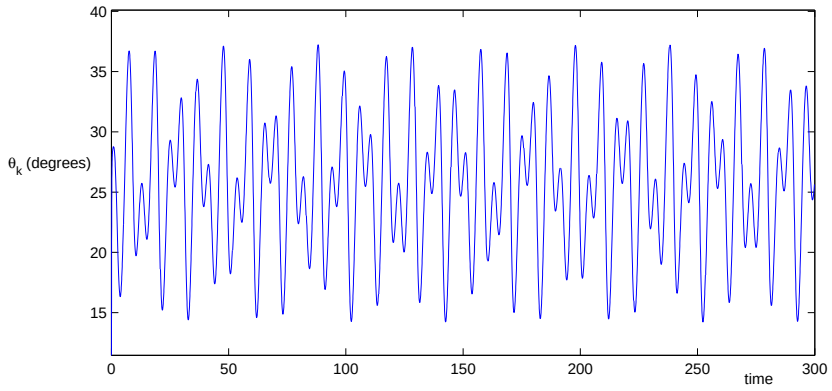


Figure 3.9: Disorderly-looking quasi-periodic motion of an agent k coupled to three agents which belong to three rings with irrational velocity ratios: $\frac{d}{dt}\theta_k = \sum_{j=1}^{3} \sin(\theta_j - \theta_k)$ where θ_1 belongs to a regularly spaced *undirected* ring, $\frac{d}{dt}\theta_1 = 0$; θ_2 is in classical cyclic pursuit with 5 other agents, $\frac{d}{dt}\theta_2 = \sqrt{3}$; θ_3 is in classical cyclic pursuit with 11 other agents, $\frac{d}{dt}\theta_3 = 1$.

switch to graph $\mathbb{G}_2$, which is a ring graph with agents interconnected in a different order: when consecutive agents in $\mathbb{G}_1$ are separated by $\frac{2\pi}{N}$, consecutive agents in $\mathbb{G}_2$ are separated by $\psi > \pi/2$ (see Figure 3.11.(c)). This means that the agents are now close to an *unstable* configuration of (3.6) or (3.12) with $\mathbb{G}_2$, and so are driven away from it (see Figure 3.11.(d)), in the direction of the initial configuration. Switching back to $\mathbb{G}_1$, (something close to) the initial situation is obtained again. Keeping switching back and forth between $\mathbb{G}_1$ and $\mathbb{G}_2$ perpetuates this cyclic behavior of relative positions, being alternatively driven towards and away from the splay state. ◇

The diversified, poorly characterized behavior of (3.6) or (3.12) with directed and time-varying $\mathbb{G}$ is in strong contrast with the behavior of the consensus algorithm on vector spaces, which is fully characterized by Proposition 3.1.1. In addition, Proposition 3.1.1 can be extended to the case where time delays are present along the communication links (see [104]), while the behavior of (3.6) or (3.12) under time delays is still under investigation even for fixed undirected $\mathbb{G}$. Already for the complete graph, delays may lead to stable synchronized solutions, stable "spread" solutions, as well as periodic oscillations [150].

3.4.3 State-dependent graphs

In some coordination problems, like the Vicsek model [148] described in Section 3.2.1 for instance, the presence of a link between two agents depends on their states. Then graph $\mathbb{G}$ is said to be *state-dependent*. Although the study of state-dependent graphs is beyond the scope of the present work, a few words about them are appropriate for completeness.

The problem of coordination under state-dependent graphs can be formulated as follows. The individual agents' control laws remain unchanged with respect to the general case (for instance, (3.12) on the circle) but the interconnections among agents, instead of being externally imposed in some way, depend on the states of the agents (for instance, $\mathbb{G}$ may depend on the relative positions $\theta_j - \theta_k$ of the agents on the circle). A frequently encountered case is

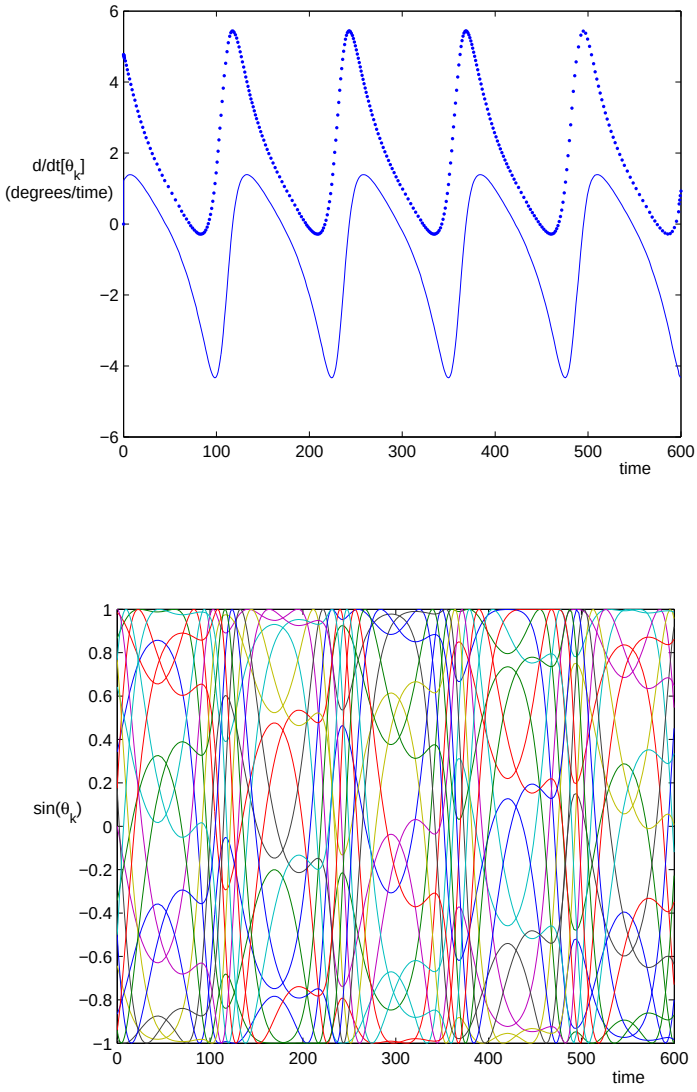


Figure 3.10: Motion of agents applying (3.12) with fixed directed coupling, such that they all periodically revert their direction of motion. Set A has 9 agents regularly spaced by $\frac{2\pi}{9}$ on the circle. Set B has 9 agents, also regularly spaced, but initially rotated by $\frac{\pi}{18}$ with respect to A. In A, $\frac{d}{dt}\theta_k = 0.04\,\sin(\theta_j - \theta_k) + 0.05\sin(\theta_l - \theta_k)$ where j is the agent of A for which $\theta_j - \theta_k = \frac{-2\pi}{9}$, and l is the agent of B for which initially $\theta_l - \theta_k = \frac{7\pi}{18}$. In B, $\frac{d}{dt}\theta_k = 0.07\,\sin(\theta_j - \theta_k) + 0.05\sin(\theta_l - \theta_k)$ where j is the agent of B for which $\theta_j - \theta_k = \frac{2\pi}{9}$ and l is the agent of A for which initially $\theta_l - \theta_k = \frac{5\pi}{18}$. Top: angular velocities of the two sets (continuous curve for A, dotted curve for B). Bottom: evolution of $\sin(\theta_k)$ for all agents k.

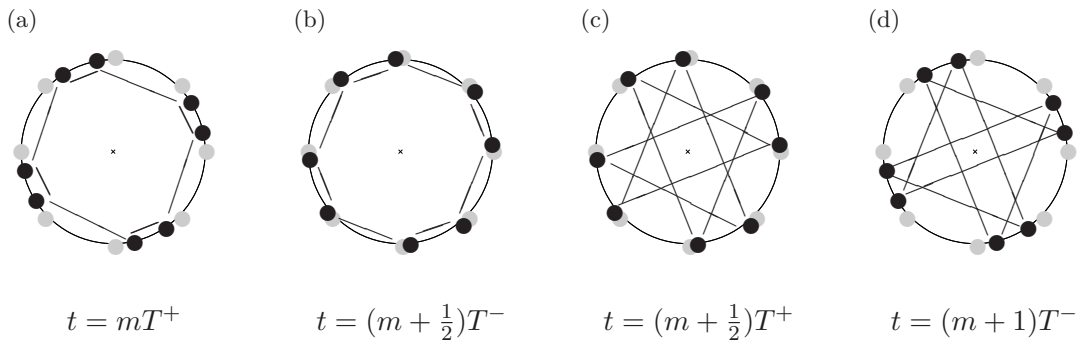


Figure 3.11: Illustration of limit cycle behavior of (3.6) or (3.12) for a varying undirected graph with $N = 8$ agents, depicted in black; $\phi = \frac{\pi}{4}$ for $\mathbb{G}_1$ and $\psi = \frac{3\pi}{4}$ for $\mathbb{G}_2$. Notations τ^- and τ^+ indicate the situation respectively just before and just after switching the graphs at time τ; T is the period of the cycle and m can take any integer value. The splay state equilibrium is depicted in light gray for reference.

the "limited sensing range" graph, where agents are interconnected if and only if their state variables are sufficiently close, i.e.

$$a_{jk} = \begin{cases} 1 & \text{if } d(x_k, x_j) \leq R \\ 0 & \text{if } d(x_k, x_j) > R \end{cases}$$

for some parameter $R > 0$ and distance measure $d(\cdot, \cdot)$. The distance measure $d(\cdot, \cdot)$ can a priori involve any "states" x_k of the agents, i.e. not necessarily exactly the variables of the synchronization algorithm. In the Vicsek model for instance, a synchronization algorithm is applied to the headings θ_k, but the graph depends on relative positions in the plane $\|r_j - r_k\|$.

A main difficulty for the analysis of coordination under state-dependent graphs is the interplay between the dynamics of the continuous state variables and the discrete graph. Usually, authors make assumptions on graph properties because it would be too difficult to analyze the state-dependent case. Results in [95, 143, 56] on vector spaces and in the present dissertation guarantee convergence *if the graph* $\mathbb{G}$ *satisfies a connectedness assumption.* In the context of state-dependent graphs, currently no tool has been proposed to check this assumption without actually solving the system. Nevertheless, it seems that progress has been made on the topic of state-dependent graphs in recent years.

An intermediate type of "state-dependent graph" is studied in [32]. The setting is a consensus algorithm on velocities in a vector space with a graph where all agents are always interconnected, but where the *weights of the links depend on the relative positions* of the agents in the vector space. It is essentially shown that, for weights a_{jk} asymptotically decreasing to zero as $\frac{1}{\|x_j - x_k\|^\beta}$ when $\|x_j - x_k\|$ tends to infinity, synchronization of the velocities is ensured for all initial conditions if $\beta < \frac{1}{2}$, while conditions on initial positions and velocities must be imposed if $\beta \geq \frac{1}{2}$.

Some results are also available for "fully" state-dependent graphs, where links between agents can vanish at finite distances.

In a design context, several authors have proposed particular, moderately complex control algorithms with the goal that, for a swarm of agents whose state-dependent interaction graph $\mathbb{G}$ is initially connected, $\mathbb{G}$ remains connected throughout the agents' motion; see [30, 77, 101] for instance.

In contrast, analysis of simple coordination algorithms with state-dependent graphs remains more elusive. In vector spaces, it is clear that for some initial conditions, the interaction graph remains connected and agents synchronize, while for others they separate into different groups that never connect again. See for instance the recent results in $\mathbb{R}$ of [3] on Kuramoto-like but limited attraction profiles, and of [14, 50] on Krause's opinion dynamics model [69]. Recently, J. Hendrickx (private communication) has been developing a result about convergence of algorithm (3.12) on S^1 where the neighbors of agent k are defined as those that are closer to θ_k than some constant R. Unfortunately, it is currently not known if and how these results might be extended to slightly more complex models, like the Vicsek model (3.8),(3.9); some restrictive conditions for convergence of the latter, as well as counterexamples similar to those of Section 3.4.2, are provided in [78]. It seems that a general and efficient tool for the analysis of dynamical systems with state-dependent graphs is still missing.

O.Q.: The question of characterizing all possible behaviors of the Vicsek model (3.8),(3.9) for general initial conditions is currently open.

From another analysis viewpoint, [87, 88] focus on the evolution of the graph itself in the framework of state-dependent graphs.

To conclude, it must be said that the present section only contains a restricted selection of examples to illustrate encountered difficulties. The reader will probably be able to find simple examples for variants of several presented properties. For instance, as a rather straightforward alternative to Example 3.4.7, the reader can try to prove that an undirected graph with constant edges, but varying weights, can lead to cycle-like behavior. Regarding Section 3.4.3, it is not too difficult to find limit cycles for the Vicsek model (3.8),(3.9).

Chapter 4

Algorithms for global synchronization on the circle

Section 3.4 highlights that synchronization on the circle is not as simple as on vector spaces. In a control framework, a natural question is then how the update rules (3.6) and (3.12) can be modified to obtain better synchronization behavior. The present chapter develops three alternatives for that purpose. The algorithm of Section 4.1 proposes to replace the sine in (3.12) by another function, in order to eliminate stable equilibria different from synchronization for fixed undirected graphs. Section 4.2 introduces randomness in the update law to break the circular symmetry; advantageous convergence results are obtained, but only in probabilistic terms and possibly extremely slowly. Section 4.3 "cheats" the non-convexity of S^1 by first synchronizing auxiliary variables in the plane, and then driving the agents to the projection on the circle of the auxiliary consensus value; this allows to essentially recover the convergence properties of vector spaces.

4.1 Modified interaction profile for fixed undirected graphs

It is an annoying fact that, for some fixed undirected graphs, the synchronization algorithms (3.6) and (3.12) have stable equilibria different from synchronization. The present section designs an alternative gradient algorithm for which the only stable equilibrium is synchronization. A continuous-time setting is chosen for convenience; a similar result could be developed in discrete-time. For simplicity, $\mathbb{G}$ is assumed unweighted.

The circle being a compact configuration space, any potential will have at least a maximum and a minimum on $(S^1)^N$, so preventing any unstable equilibrium is not possible[1].

4.1.1 General idea

For $\mathbb{G}$ fixed and undirected, (3.12) is a gradient system for V_θ defined in (3.10). Therefore a system applying (3.12) always converges to a set of equilibria, and the stability of these equilibria can be determined by examining the Hessian of V_θ: an equilibrium is stable if

[1]This is true for a larger class of dynamical systems than just gradient algorithms, and for a larger class of manifolds than the sphere: a theorem of Milnor [90] implies that, if a system admits a continuous feedback that globally asymptotically stabilizes it to a point, then the state space of that system is diffeomorphic to a vector space.

and only if the Hessian is positive semidefinite, the only zero eigenvalue corresponding to a uniform motion of the swarm $\theta_k = \theta_k + a \ \forall k \in \mathcal{V}$.

Definition 4.1.1: *Consider a continuous-time synchronization algorithm on S^1 of the form*

$$\tfrac{d}{dt}\theta_k = \sum_{j \rightsquigarrow k} f(\theta_j - \theta_k) \ , \quad k = 1, 2, ..., N \, .$$

The function $f : S^1 \to \mathbb{R}$ is called the interaction profile*.*

The properties of the Hessian at a particular equilibrium are directly linked to the interaction profile $f(\theta)$. In Example 3.4.2 with the ring graph, equilibria are stable if interconnected agents are closer than $\frac{\pi}{2}$, for which $f(\theta)$ has a positive slope, and unstable if they are further apart than $\frac{\pi}{2}$, for which $f(\theta)$ has a negative slope. From the arguments in Example 3.4.2, if the slope of $f(\theta)$ was only positive up to $\frac{\pi}{a}$ for some $a > 2$, then agents would have to be closer than $\frac{\pi}{a}$ at a stable equilibrium for the ring graph. If a is sufficiently large with respect to N, then it will not be possible to distribute the agents on the circle as on Figure 3.5, so there will be no stable balanced equilibrium anymore.

4.1.2 Algorithm and stability proof

Assume (a bound on) the number N of agents in the swarm is available to each agent. Define the interaction profile

$$g(\theta) = \begin{cases} \frac{-a}{N-1}(\pi + \theta) & \text{for } \theta \in [-\pi, -\frac{\pi}{N}] \\ a\,\theta & \text{for } \theta \in [-\frac{\pi}{N}, \frac{\pi}{N}] \\ \frac{a}{N-1}(\pi - \theta) & \text{for } \theta \in [\frac{\pi}{N}, \pi] \end{cases} \tag{4.1}$$

for some $a > 0$, extended 2π-periodically outside the above intervals, as represented on Figure 4.1. Replacing the sinusoidal interaction profile of (3.12) by $g(\theta)$ defines an alternative continuous-time synchronization algorithm

$$\tfrac{d}{dt}\theta_k = \sum_{j \rightsquigarrow k} g(\theta_j - \theta_k) \ , \quad k = 1, 2, ..., N \tag{4.2}$$

which satifies all invariance and communication constraints.

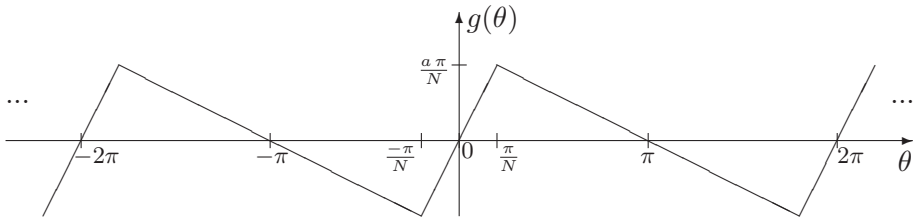


Figure 4.1: The alternative interaction profile $g(\theta)$.

Function $g(\theta)$ in (4.1) is the gradient of

$$(z(\theta))^2 = \begin{cases} \frac{a\pi^2}{2N(N-1)} + \frac{a}{N-1}(-\pi\,\theta - \frac{\theta^2}{2}) & \text{for } \theta \in [-\pi,\, -\frac{\pi}{N}] \\ \frac{a}{2}\theta^2 & \text{for } \theta \in [-\frac{\pi}{N},\, \frac{\pi}{N}] \\ \frac{a\pi^2}{2N(N-1)} + \frac{a}{N-1}(\pi\,\theta - \frac{\theta^2}{2}) & \text{for } \theta \in [\frac{\pi}{N},\, \pi] \end{cases}$$

extended 2π-periodically outside the above intervals. Function $|z(\theta)|$ is even, has a minimum for $\theta = 0$, a maximum for $\theta = \pi$ and evolves monotonically and continuously in between, similarly to the sinusoidal distance measure $2|\sin(\frac{\theta}{2})|$. Algorithm (4.2) is a gradient descent algorithm for the cost function

$$V_g = \tfrac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\,(z(\theta_j - \theta_k))^2 \ . \tag{4.3}$$

The following proves that synchronization is the only stable solution for a swarm applying (4.2) by examining the Hessian H_g of V_g. V_g is not twice continuously differentiable everywhere, such that its Hessian is not properly defined for agents exactly separated by $\frac{\pi}{N}$. This problem can be circumvented by smoothing the edges of $g(\theta)$, which changes nothing to the general argument.

Proposition 4.1.2: *Consider a swarm of N agents, interconnected according to a connected fixed undirected graph $\mathbb{G}$, that evolve on the circle by applying algorithm (4.2). The agents always converge to the set of equilibria corresponding to the critical points of V_g. Moreover, the only asymptotically stable equilibrium is synchronization.*

Proof: The fact that (4.2) is a gradient system for V_g directly implies that the agents always converge to a set of critical points of V_g, as for Proposition 3.3.2 in the sinusoidal case. Synchronization, as the global minimum of V_g, is a stable equilibrium. The following steps prove that no other stable equilibria can exist: (i) characterize the Hessian H_g of V_g at the equilibrium points; (ii) link the eigenvalues of H_g to the eigenvalues of a Laplacian matrix; (iii) show that if the Hessian is positive semidefinite with 0 eigenvalue only in the direction of uniform motion, then the graph $\mathbb{G}_p$, containing edge $\{j,k\}$ if and only if j and k are closer than $\frac{\pi}{N}$, must be connected. This concludes the proof: it implies that at a stable equilibrium, all agents must be within a semicircle; but like for the sinusoidal profile, with all agents within a semicircle, the convexity properties of vector space consensus algorithms are retrieved so there can be no other equilibrium than synchronization.

(i) By definition, element j,k of the Hessian matrix H_g equals $(h_g)_{jk} = \frac{\partial^2 V_g}{\partial\theta_j \partial\theta_k}$. Since $\frac{\partial V_g}{\partial\theta_k} = -\sum_{j\rightsquigarrow k} g(\theta_j - \theta_k)$, this yields

$$(h_g)_{jk} = \begin{cases} 0 & \text{if } j \text{ and } k \text{ are not connected} \\ -a & \text{if } j \text{ and } k \text{ are connected and closer than } \frac{\pi}{N} \\ b := \frac{a}{N-1} & \text{if } j \text{ and } k \text{ are connected and further apart than } \frac{\pi}{N} \\ -\sum_{j=1\,,\,j\neq k}^{N} (h_g)_{jk} & \text{for diagonal elements } j = k \end{cases}$$

This is true for any values of the θ_k, $k = 1, 2, ..., N$, in particular at equilibria.

(ii) H_g is a symmetric matrix with zero row and column sums. Thus it has an eigenvalue 0 associated to eigenvector $\mathbf{1}_N$, which is the direction of a uniform translation ($\theta_k \to \theta_k + a$ $\forall k \in \mathcal{V}$ for some arbitrary $a \in \mathbb{R}$). Denote the other eigenvalues of H_g by $\lambda_2 \leq \lambda_3 \leq ... \leq \lambda_N$ without loss of generality.

Define $H_1 = H_g - bM_1$ where $M_1 \in \mathbb{R}^{N\times N}$ is the matrix whose elements are all equal to one. All off-diagonal elements of H_1 are non-positive and its rows and columns sum to $-Nb$. M_1 has eigenvalue N associated to eigenvector $\mathbf{1}_N$ and eigenvalue 0 with multiplicity $N-1$ associated to the remaining subspace. Therefore, the eigenvalues of H_1 are $-Nb$, λ_2, $\lambda_3, ..., \lambda_N$. Now define $H_2 = H_1 + Nb\, I_N$. Matrix H_2 has non-positive off-diagonal elements, which can take values

$$(h_2)_{jk} = \begin{cases} -(a+b) & \text{if } (h_g)_{jk} = -a \\ -b & \text{if } (h_g)_{jk} = 0 \\ 0 & \text{if } (h_g)_{jk} = b \end{cases} .$$

Moreover, it is symmetric and has zero row and column sums. Thus it can be seen as the Laplacian of an undirected weighted graph $\mathbb{G}_2$ on N vertices, with two possible weights $(a+b)$ and b. Obviously, the eigenvalues of H_2 are 0, $(\lambda_2 + Nb)$, $(\lambda_3 + Nb)$,... , $(\lambda_N + Nb)$.

(iii) Assume that $\lambda_2 > 0$, i.e. the swarm is at an asymptotically stable equilibrium, or $\lambda_2 = 0$. Then the second eigenvalue of H_2 must be larger or equal to Nb. Define the graph $\mathbb{G}_p$ on N vertices and containing the edges that have weight $(a+b)$ in $\mathbb{G}_2$. The goal is to show that if $\mathbb{G}_p$ is not connected, then the second-smallest eigenvalue of H_2 is smaller than Nb. Indeed, this implies conversely that if $\lambda_2 \geq 0$, then the graph $\mathbb{G}_p$, containing edge $\{j,k\}$ if and only if j and k are closer than $\frac{\pi}{N}$, must be connected.

The eigenvalues of a symmetric matrix H can be computed with the Rayleigh quotient $r(x) = \frac{x^T H x}{x^T x}$ for $x \in \mathbb{R}^N$. The smallest eigenvalue is the minimum of $r(x)$ over $\mathbb{R}^N$ and the corresponding vector x_* is the associated eigenvector; the second smallest eigenvalue is the minimum of $r(x)$ for x in the hyperspace orthogonal to x_*. Assume that $\mathbb{G}_p$ is not connected. Renumber the agents such that the agent set $\{1, 2, ..., m\}$ is disconnected from the agent set $\{m+1, m+2, ..., N\}$ in $\mathbb{G}_p$, for some $m \in \{1, 2, ..., N-1\}$. Then the renumbered H_2 can be put in block form

$$H_2 = \begin{pmatrix} H_{d1} & H_{od} \\ H_{od}^T & H_{d2} \end{pmatrix} \quad \text{where } H_{d1} \in \mathbb{R}^{m\times m}\,,\ H_{d2} \in \mathbb{R}^{(N-m)\times(N-m)}\,,\ H_{od} \in \mathbb{R}^{m\times(N-m)}\,,$$

and H_{od} contains no element of value $-(a+b)$. Now consider the vector $\mathbf{v}_*$ whose first m elements are $\frac{1}{m}$ and whose remaining elements are $\frac{-1}{(N-m)}$. Then $\mathbf{1}_N^T \mathbf{v}_* = 0$, so $\mathbf{v}_*$ is a candidate vector to define the second smallest eigenvalue of H_2. Compute $\mathbf{v}_*^T \mathbf{v}_* = (\frac{1}{m} + \frac{1}{N-m})$. Compute

$$r(\mathbf{v}_*) = \frac{\mathbf{v}_*^T H_2 \mathbf{v}_*}{(\frac{1}{m} + \frac{1}{N-m})} = -\mathbf{v}_*^T \begin{pmatrix} H_{od}\,\mathbf{1}_{N-m} \\ -H_{od}^T\,\mathbf{1}_m \end{pmatrix} = -(\tfrac{1}{m} + \tfrac{1}{N-m})\ \mathbf{1}_m^T H_{od}\,\mathbf{1}_{N-m}\,.$$

Matrix H_{od} contains elements $-b$ or 0, and must contain at least one 0 if the initial graph $\mathbb{G}$ is connected. Therefore

$$r(\mathbf{v}_*) < (\tfrac{1}{m} + \tfrac{1}{N-m}) m(N-m)b = Nb$$

so the second-largest eigenvalue of H_2 is smaller than Nb. □

Algorithm (4.2) solves the problem of local minima in V_θ. However, it maybe introduces numerous unstable equilibria. Moreover, for varying and directed graphs, the behavior of (4.2) is not better characterized than for (3.12): convergence can only be ensured if the agents are located within a semicircle, and then they always converge to synchronization. The algorithms proposed in the following two sections aim at establishing better convergence properties for possibly varying and directed graphs.

4.2 Introducing randomness in link selection

The present section introduces a so-called "Gossip Algorithm" (see [17] and references therein) in order to improve synchronization behavior on the circle. It is described in discrete-time for easier formulation. The idea is to keep the update law (3.6), but at each time randomly select at most one of the in-neighbors in $\mathbb{G}(t)$ for each agent. The hope that reducing interconnection links could be successful comes from the observation that (3.6) and (3.12) have nice convergence properties when $\mathbb{G}$ is a tree. Another argument is that some time-varying randomness in the selection of neighbors should destroy the stability of spurious local minima of V_θ because synchronization is the only common minimum over all the partial graphs. The directed version of the Gossip Algorithm is presented with an extreme behavior where an agent directly jumps to the position of the selected neighbor. This algorithm has the advantage of being completely independent of the underlying configuration space and can in fact be applied to reach agreement starting from any set of N initial symbols. A more moderate version of the directed Gossip Algorithm is mentioned without much analysis, as it behaves similarly to the undirected version.

4.2.1 Algorithm description

The idea is that each agent k selects at most one of its in-neighbors $j \rightsquigarrow k$ in its update law at each time. However, in order to satisfy the equivalence of all agents, k may not privilege any of its neighbors — it is just allowed, for weighted $\mathbb{G}$, to take the different weights of the corresponding edges into account. Always choosing the neighbor with maximum weight could disconnect the interaction graph. Therefore, it is necessary to select the retained neighbor *randomly* among the $j \rightsquigarrow k$. A natural probability distribution for neighbor selection would follow the weights of the edges. Convergence towards synchronization is actually easier to achieve with updates along directed edges, but since some situations may require to retain the symmetry of undirected edges, algorithms are proposed for both cases.

Gossip algorithm (directed). See Figure 4.2.(a) and (b). At each update t,

1. each agent k randomly selects an agent $j \rightsquigarrow k$ with probability $a_{jk} / \left(\beta + \sum_{l \rightsquigarrow k} a_{lk}\right)$, where $\beta > 0$ is the weight for choosing no agent;
2. $\theta_k(t+1) = \theta_j(t)$ if agent k chooses neighbor j at time t, and $\theta_k(t+1) = \theta_k(t)$ if it chooses no neighbor.

Gossip algorithm (undirected). See Figure 4.2.(a) and (c). At each update t,

1. each agent k randomly selects one neighbor or none, as in the directed case;
2. if k chooses j `AND` j chooses k at time t, then k and j move towards the midpoint of the shortest arc between them, i.e. $\theta_k(t+1) = \theta_j(t+1) = \arg(e^{i\theta_k(t)} + e^{i\theta_j(t)})$. If k chooses no neighbor or a neighbor j which does not choose k, then $\theta_k(t+1) = \theta_k(t)$.

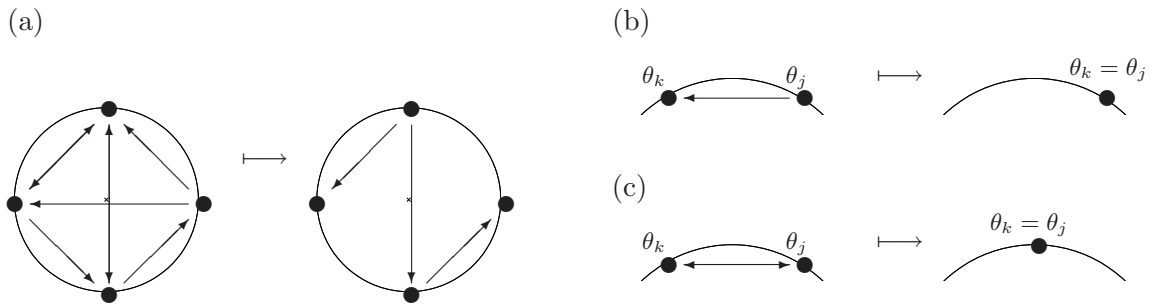


Figure 4.2: Update laws for the Gossip Algorithm: (a) For both variants (directed and undirected), agents start by selecting at most one of their in-neighbors. (b) For the directed Gossip Algorithm, agent k directly jumps to the position of its chosen in-neighbor j (or does not move). (c) For the undirected Gossip Algorithm, agents k and j, choosing each other, move towards the midpoint of the shortest arc between them (or else do not move).

With the proposed edge selection procedure, the neighbor chosen at time $t+1$ is independent of the neighbor chosen at time t (up to, for varying graphs, a possible dependence of $\mathbb{G}(t+1)$ on $\mathbb{G}(t)$).

The directed Gossip Algorithm proposed above is extreme in the sense that agent k directly jumps to the position of its selected neighbor. A more moderate directed Gossip Algorithm would apply the update law $\theta_k(t+1) = \arg(\mu\, e^{i\theta_k(t)} + e^{i\theta_j(t)})$ with inertia $\mu > 0$.

The authors of [17] perform a detailed analysis of an undirected Gossip Algorithm for synchronization in vector spaces. Since convergence towards synchronization is always ensured on vector spaces, the problem is to quantify the convergence *rate* as a function of the interconnection graph and probability (i.e. weights) distribution. On the circle, convergence towards synchronization is not obvious a priori. It is analyzed in the following section.

4.2.2 Convergence analysis

Definition 4.2.1: *In the present stochastic setting, the N agents are said to* asymptotically converge towards synchronization with probability 1 *if for any initial condition, for any $\varepsilon > 0$ and $\kappa \in (0,1)$, there exists a time T after which the maximal distance $|\theta_k(T) - \theta_j(T)|$ between any pair of agents is smaller than ε with probability larger than κ.*

The strategy to prove asymptotic convergence towards synchronization with probability 1 in different settings is based on the following facts, valid for time-varying and directed interconnection graphs.

Lemma 4.2.2: *If all agents are located within an open semicircle, asymptotic synchronization with probability 1 is ensured with the Gossip Algorithms on δ-digraphs if there exists a finite horizon T and a probability $p_0 > 0$ such that $\forall t$, the graph formed by the links selected during $[t, t+T]$ is root-connected with probability p_0 at least.*

Proof: One realization of a Gossip Algorithm can be viewed as a deterministic algorithm for a particular time-varying graph (featuring at most one link per agent at a time). According to Proposition 3.1.1, a sufficient condition for exponential synchronization on the semicircle

is that this graph is uniformly connected.

For a particular ε and initial condition with all agents within a semicircle, denote by $N_\varepsilon T_0$, with N_ε integer, a time after which all agents are necessarily in an interval smaller than ε, for all possible δ-digraphs that are uniformly connected over T_0; by time-scale symmetry and since convergence on the semicircle is exponential, N_ε is independent of T_0 and finite. It remains to prove that $\forall \kappa \in (0,1)$, there exists a T_0 such that the graph formed by the edges selected in the Gossip Algorithm is root-connected during the first N_ε consecutive T_0-time spans with probability larger than κ.

If a graph is root-connected on T with probability p_0, then it is root-connected on $T_0 = mT$ with probability at least $p_1 = (1-(1-p_0)^{\lfloor m \rfloor})$ which can be made arbitrarily close to 1 with a sufficiently large m. The graph is then root-connected on the first N_ε consecutive T_0-time spans with probability at least $p_2 = p_1^{N_\varepsilon}$. For any ε and initial condition, it suffices to select m such that $p_2 > \kappa$. □

The problem of global synchronization can thus be reduced to the study of "bringing all agents within a semicircle". Indeed, write $\kappa = \kappa_1 \kappa_2$ with $\kappa_1 < 1$ and $\kappa_2 < 1$. If it can be ensured for any $\kappa_1 \in (0,1)$ that all agents are within a semicircle after a finite time T_{κ_1} with probability κ_1, then applying Lemma 4.2.2 with κ replaced by κ_2 ensures asymptotic synchronization with probability 1.

Lemma 4.2.3: *Consider a sequence σ of link selections over a finite time span T_σ, whose probability to appear at least once in a time span $[t, t+T_s)$ is at least $p_{\sigma s} > 0$ $\forall t$ and for some finite $T_s \geq T_\sigma$. If applying a Gossip Algorithm with sequence σ ensures that all agents end up in a semicircle* for all initial conditions*, then for any $\kappa \in (0,1)$ there exists a finite time T_h after which the Gossip Algorithm (with random sequence) has driven all the agents within a semicircle with probability $p > \kappa$.*

Proof: If σ appears with probability $p_{\sigma s}$ in any time span of length T_s, then it appears at least once in any time span of length $T_h = mT_s$ with probability at least $(1-(1-p_{\sigma s})^{\lfloor m \rfloor})$, which can be made $> \kappa$ by taking m sufficiently large. Thus for m sufficiently large, there is probability $p_3 > \kappa$ that a link sequence appears during T_h such that all agents are within a semicircle at the end of that link sequence; after the sequence, convexity arguments ensure that the agents never leave this semicircle. □

A finite sequence σ of link selections driving all agents within a semicircle, regardless of the initial condition, is called a *synchronizing sequence* in the following. Thanks to Lemma 4.2.2 and Lemma 4.2.3, the study of asymptotic synchronization with probability 1 is reduced to the search for synchronizing sequences that appear in every time span of some bounded length T_s with probability $p \geq p_s$ for some fixed $p_s > 0$. The following results identify such sequences for both types of Gossip Algorithms.

Proposition 4.2.4: *Consider a set of N agents interconnected according to a uniformly connected δ-digraph $\mathbb{G}$. If the agents apply the directed Gossip Algorithm, with a fixed finite $\beta > 0$, then they asymptotically synchronize with probability 1.*

Proof: The proof is based on the construction of a synchronizing sequence appearing with probability at least p_s in some time span T_s. Semicircle arguments, such as Lemma 4.2.2, are in fact not needed.

Consider that $\mathbb{G}$ is uniformly connected over T_c, and denote by δ and Δ the minimum and maximum values of all $a_{jk} \neq 0$ for all t. The links appearing in $[t_0+nT_c, t_0+(n+1)T_c)$ contain a rooted tree for $n = 0, 1, 2...$. Since there are N possible roots, over $[t_0, t_0+N(N-1)T_c)$, there exists an agent r serving as a root for at least $N-1$ trees, called $tr_1, tr_2, ...tr_{N-1}$, appearing in non-ovelapping time intervals. Consider the following link sequence σ. Over time intervals not corresponding to tr_n, $n = 1...N-1$, no agent chooses a neighbor (no updates). Over the time interval of tr_1, a child c of r chooses its link with r, such that $\theta_c(t+1) = \theta_r(t)$, and all other agents choose no neighbor[2]. Over the time intervals of further trees tr_n, $n = 2...N-1$, choose just one update $\theta_j(t+1) = \theta_k(t)$, where k is either r or a previously updated agent and j is any of k's children in tr_n that was not updated yet; no other agents move. One easily checks that this is always possible until all agents have been updated and are thus located at the initial position of agent r. It remains to show that this link sequence appears with finite probability p_s in a time span T_s.

At any time t, the probability for agent k to choose a particular link is at least $\frac{\delta}{(N-1)\Delta+\beta} =: \eta_1 > 0$, while the probability of k choosing no link is at least $\frac{\beta}{(N-1)\Delta+\beta} =: \eta_2 > 0$. Thus the probability to choose a particular link *and no other link in the graph* is at least $\eta_1\,\eta_2^{N-1}$ and the probability to choose no link at all is η_2^N. Therefore, the probability to choose the particular sequence σ in time span $T_s = N(N-1)T_c$ is at least $p_s = (\eta_1\,\eta_2^{N-1})^{N-1}\,(\eta_2^N)^{N(N-1)T_c-(N-1)} > 0$ which is finite (although potentially very small, and underestimated). □

The directed Gossip Algorithm without inertia is a very particular case. The convergence proof for the moderate version of the directed Gossip Algorithm, with inertia $\mu > 0$, as well as for a related continuous-time directed Gossip Algorithm, can be developed along the lines of the proof for the undirected Gossip Algorithm.

Proposition 4.2.5: *Consider a set of N agents interconnected according to an undirected uniformly connected δ-digraph $\mathbb{G}$. If the agents apply the Gossip Algorithm for undirected graphs, with a fixed finite $\beta > 0$, then they asymptotically synchronize with probability 1.*

Proof: When $\mathbb{G}$ is uniformly connected, it suffices to bring all the agents within a semicircle to satisfy the conditions of Lemma 4.2.2. The remainder of the proof is based on the construction of a synchronizing sequence appearing with probability at least p_s in some time span T_s. This is more difficult to achieve than in the directed case. Denote by δ and Δ the minimum and maximum values of all $a_{jk} \neq 0$ in $\mathbb{G}(t)$ over all j, k and all t.

First assume that $\mathbb{G}$ is *time-invariant* and connected. Then $\mathbb{G}$ contains an undirected spanning tree $\mathcal{S}_N$. The synchronizing sequence for this tree is built iteratively from a synchronizing sequence for smaller trees.
Denote by $\mathcal{S}_q$ a sub-tree of $\mathcal{S}_N$ containing $q < N$ vertices. Suppose that a link sequence σ_q is known to bring the q agents of the partly constructed tree $\mathcal{S}_q$ within an arc of length $q\frac{\alpha}{N}$, with $\alpha \in (0, \pi/2)$, for any initial condition. To initialize the sequence, σ_2 just contains a single link, two agents average their values and are thus within $0 < \frac{2\alpha}{N}$. With slight abuse of notation, $\mathcal{S}_q$ is also used to denote the vertex set of the tree $\mathcal{S}_q$. Denote by k a new agent to add and by j the agent to which it will be connected in order to build a larger sub-tree $\mathcal{S}_{q+1}$. The link sequence $\sigma_{q+1} = \sigma_q\,;\,\{j \leftrightsquigarrow k\}\,;\,\sigma_q\,;\,\{j \leftrightsquigarrow k\}\,;\,\sigma_q$; ... (repeat $x > \log_2(\pi N/\alpha)$ times)

[2]Although it could be possible to choose further links for tr_1, this is the minimum achievable for any graph, namely when r has only one child c in tr_1 and the link $r \rightsquigarrow c$ appears at the end of the time interval of tr_1.

brings the $q+1$ agents within an arc of length $(q+1)\frac{\alpha}{N}$, for any initial condition. Indeed, for any position of the agents, denote by γ the distance from k to the furthest point of the arc containing $\{\theta_i : i \in \mathcal{S}_q\}$, along the direction of the shortest arc between k and j. Then apply the first sequence σ_q to get a first value γ_0 of γ, satisfying $\gamma_0 \leq \pi + q\frac{\alpha}{N}$. After one iteration of $\{j \rightsquigarrow k\}$; σ_q, the agents of $\mathcal{S}_q$ are within $q\frac{\alpha}{N}$ again, but $\gamma - q\frac{\alpha}{N} \leq (\gamma_0 - q\frac{\alpha}{N})/2$. Thus after x iterations, $\gamma - q\frac{\alpha}{N} \leq (\gamma_0 - q\frac{\alpha}{N})/(2^x) \leq \pi/(2^x) < \frac{\alpha}{N}$ such that the $q+1$ agents are within $(q+1)\frac{\alpha}{N}$.
The sequence σ_N built iteratively in this way is synchronizing, as it brings all agents within $\frac{N\alpha}{N} = \alpha < \pi/2$ for any initial condition. Moreover, it has finite length T_σ (although T_σ grows exponentially with N). It contains a single link at each time t. The probability to choose a particular link at time t is at least $\eta_3 := \eta_1^2 \eta_2^{N-2} > 0$ (the two agents to be connected choose each other mutually, the others choose no neighbor), where $\eta_1 = \frac{\delta}{(N-1)\Delta+\beta}$ and $\eta_2 = \frac{\beta}{(N-1)\Delta+\beta}$. σ is chosen at least with finite probability $(\eta_3)^{T_\sigma}$ in time span T_σ.

If $\mathbb{G}$ is time-varying, then for an arbitrarily large B, choosing a long enough time span $T_B \geq BT_c$ ensures that there is an undirected tree tr_1 appearing on B non-overlapping time spans of length T_c during T_B. Then the synchronizing sequence σ_N (built above) of the fixed tr_1 can be implemented by choosing the next link in the sequence if it is available, and no link else; if $B \geq T_\sigma$, the whole sequence can be applied, leaving the agents in the same configuration as with constant $\mathbb{G}$. The probability of this sequence is then at least $(\eta_3)^{T_\sigma}(\eta_2^N)^{T_B - T_\sigma}$, which can be small but remains finite. □

4.2.3 Simulation results and convergence rate

Propositions 4.2.4 and 4.2.5 prove that a set of agents applying the Gossip Algorithms globally asympotically synchronize with probability 1, but they say little about the convergence rate. From a design viewpoint, it is interesting to examine which probability distributions for the choice of edges achieve the best performance in terms of convergence rate.

Undirected Gossip Algorithm Bounds on the convergence rate can be obtained from the synchronizing sequence constructed in the proof, but they would probably be very conservative. The sequences proposed in the proofs are just examples leading to easy discussion and probability bounds. Clearly, (probably many) other synchronizing sequences can be found, increasing the probability to have completed one after some fixed time. Specific synchronizing sequences can certainly be built depending on properties of $\mathbb{G}$; however, as mentioned in Section 3.4, already the characterization of globally S^1-synchronizing fixed undirected graphs is currently an open question. The convergence rate should also take into account sequences that synchronize the agents for particular initial conditions. In summary, finding useful bounds to assess the convergence rate of the Gossip Algorithms is a tough open question.

For consensus on vector spaces, the question of estimating the expected convergence rate and of choosing a probability distribution that maximizes the expected convergence rate is addressed in [17]. The latter result is locally valid on the circle. However, local convergence is not the most relevant part, since the fundamental problem is to first get all agents within a semicircle; this is further illustrated on simulations, see below. Also, unlike on vector spaces, on the circle the qualitative behavior of the swarm can critically depend on initial conditions, so probability distributions on initial positions of the agents would have to be taken into account, and the different possible intermediate positions resulting from applying the control

algorithm are of importance. Overall, the choice of an optimal probability distribution seems to be really difficult to address; maybe some general rules can be deduced without requiring expressions for the convergence rate.

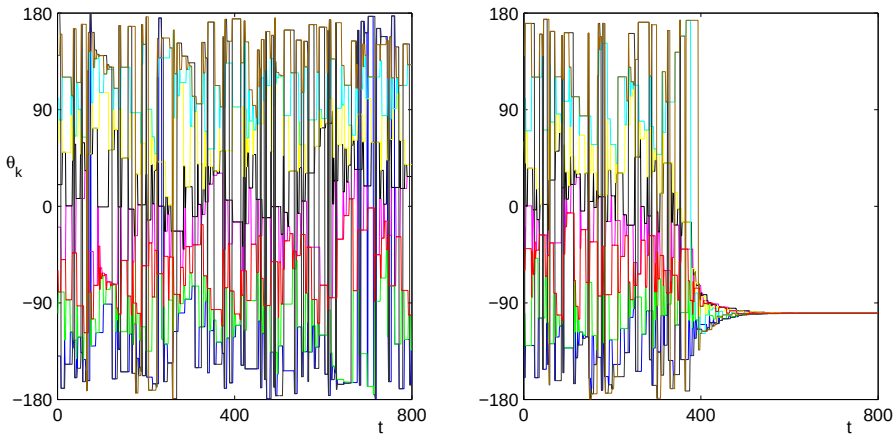


Figure 4.3: Evolution of the θ_k (in degrees) from an initial splay state, for the undirected Gossip Algorithm on an undirected cycle graph with $N = 9$ agents and probability distribution $\beta = 1$, $a_{jk} = 1\ \forall j \rightsquigarrow k$. About 75% of trials exhibit no synchronization after 800 iterations (as on the left plot). Convergence in the remaining trials is fast once agents are in a semicircle (as on the right plot).

Simulations confirm that the undirected Gossip Algorithm favors global synchronization. It is also observed that convergence can be slow if the initial condition is (close to) a local minimum of V_θ different from synchronization. As an illustration, consider a set of N agents connected by a fixed undirected ring graph and the initial situation in splay state, with interconnected agents separated by $\frac{2\pi}{N} < \frac{\pi}{2}$, as in Example 3.4.2. For the Gossip Algorithm, this graph does not look too bad a priori: suppressing one link makes it a tree, for which synchronization is the only stable configuration, so if one link is not selected during a sufficiently long time span, the agents can be driven close to synchronization.

Figure 4.3 shows two simulations starting close to the splay state with $N = 9$, $\beta = 1$ and $a_{jk} = 1\ \forall j \rightsquigarrow k$. The simulation on the left has made no progress towards synchronization after 800 iterations; repeating simulations, this situation appears roughly 3 times out of 4, which means that in Definition 4.2.1 for this particular initial condition, $T > 800$ for $\delta = 0.25$ and any small ε. The simulation on the right shows a case where synchronization is achieved: at one point in time, the agents end up within a semicircle and from this point on convergence is much faster.

Directed Gossip Algorithm Under the directed version of the Gossip Algorithm — thanks to the extreme choice to introduce zero inertia — agents in fact jump between a discrete set of possible positions, corresponding to the initial positions of the N agents. This highlights an important property of the directed Gossip Algorithm: it can in fact be applied on *any set of symbols*. Proposition 4.2.4 ensuring asymptotic synchronization purely relies on the evolution of agents between N different "symbols", *completely independently of the*

underlying manifold. Each time a position is left empty (implying that the synchronization process progresses as an agent joins other ones), that position can never be reached again in the future; this process goes on until all agents are on the same position after a finite time. The simulation represented on Figure 4.4 clearly illustrates this behavior.

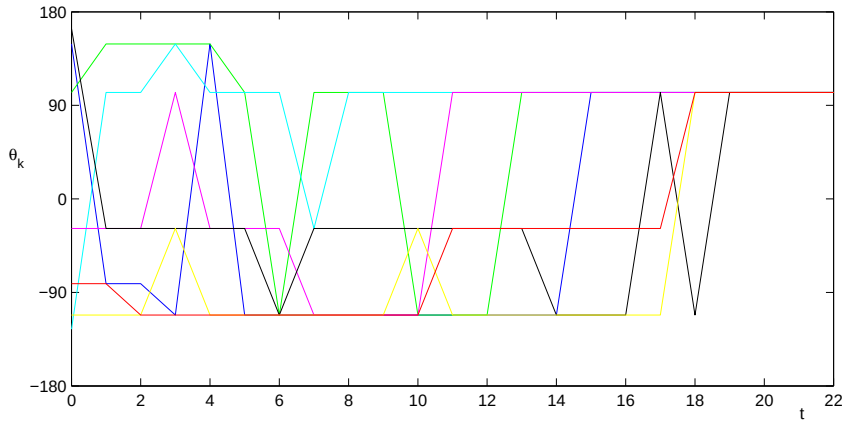


Figure 4.4: Evolution of the θ_k (in degrees) from an arbitrary initial state, for the directed Gossip Algorithm on an undirected cycle graph with $N = 7$ agents and probability distribution $\beta = 3$, $a_{jk} = 1\ \forall j \rightsquigarrow k$.

In this context, a natural measure of convergence rate is the *expected synchronization time*, i.e. the average time, over all possible link choices, after which all agents are on the same position. Since the directed Gossip Algorithm evolves between a discrete set of positions, a Markov chain framework can be applied to obtain an explicit formula for the expected synchronization time, at least for fixed graph $\mathbb{G}$. This requires to list all possible partitions of the N agents into N (possibly empty) sets and build the transition probability matrix P between all those partitions, where p_{jk} gives the probability to go from partition j to partition k. Then define P_s equal to P except that the partition corresponding to synchronization (N agents in one set, the $N-1$ other sets empty) is considered as an exit state, i.e. the transition probabilities on the row of partition s, corresponding to synchronization, are all set to $p_{sj} = 0$ $\forall j$, instead of keeping $p_{ss} = 1$. It is then easy to see that the expected synchronization time is given by

$$t_{\text{synch. average}} = \mathbf{e}_d \left(I_{P_s} - P_s\right)^{-2} P_s \, \mathbf{e}_s^T \tag{4.4}$$

where $\mathbf{e}_d$ is the initial state row vector, containing all zeros except a 1 for the "distributed partition" d in which each set contains exactly one agent; $\mathbf{e}_s$ is the row vector characterizing the synchronization partition, containing all zeros except a 1 for partition s; and I_{P_s} is the identity matrix of the same dimension as P_s.

It is important to note that this measure of convergence rate — including its numerical value — is independent of S^1 and *independent of the initial positions of the agents* (unless some agents are initially perfectly synchronized). The only remaining parameters are the graph $\mathbb{G}$ and the probability distribution. There seems to be an interesting interplay between these two parameters: in a directed rooted tree, the optimal link choice probability is clearly to set $\beta = 0$, while e.g. for a directed cycle this choice would make synchronization impossible.

The challenge is that it is not obvious for the agents — which do not know the whole graph $\mathbb{G}$ — to distinguish between these two situations based on their local information.

Unfortunately, the practical use of formula (4.4) is limited. One issue is that the size of matrix P_s grows exponentially with the number of agents. Moreover, it seems difficult to extract the influence of the link choice probability distribution from formula (4.4) and the construction of P_s. Therefore a detailed exploration of the influence of graph and probability distribution on convergence rate has not been performed in this framework. Some exploration done with small graphs ($N \leq 5$) seems to indicate that the minimum of $t_{\text{synch. average}}$ with respect to the probability distribution is very flat, but it is questionable how well this extends to larger graphs. In addition, (4.4) cannot be applied for time-varying graphs. Note that, since the directed Gossip Algorithm is independent of the underlying manifold, related questions are in fact closer to the field of graph theory than to the geometric viewpoint of the present work.

O.Q.: The question of characterizing the expected global convergence rate of the Gossip Algorithms towards synchronization is currently poorly solved. The related question of how to choose the probability distribution for random link selection in order to reach synchronization as quickly as possible is also currently open.

4.3 Dynamic controller with auxiliary variables

The present section proposes a global synchronization algorithm on the circle that (almost) recovers the convergence properties of vector spaces at the cost of introducing *auxiliary variables*: the state space of agent k is augmented by adding to its position $\theta_k \in S^1$ a variable $y_k \in \mathbb{R}^2 \cong \mathbb{C}$. The agents must not only memorize and update their auxiliary variable, but also communicate it to their out-neighbors. Thus the proposed strategy is feasible for engineering applications where real variables can be communicated along the links represented by the graph edges. It is certainly a questionable model to describe natural phenomena. The controller is called *dynamic* since, with the y_k, it contains state variables of its own.

4.3.1 Algorithm description

The basic idea of the algorithm is to "first" synchronize the auxiliary variables y_k to some point $\bar{y}$ in the plane, and "then" drive all the agents to the projection of $\bar{y}$ onto the circle; since convergence is asymptotic, those two steps are actually implemented in parallel. To simplify notations, the plane is viewed both as $\mathbb{C}$ and as $\mathbb{R}^2$, where the first component of $\mathbb{R}^2$ is identified with the real part and the second component with the imaginary part.

For synchronizing the y_k, the linear consensus algorithm (3.1),(3.3) or (3.2),(3.3) can be used respectively in continuous- or discrete-time (replacing x_k by y_k). The $y_k(0)$ can be initialized arbitrarily. For θ_k to track the projection of y_k on S^1, the cost function $V_{\text{proj}}(\theta_k) = \frac{1}{2}\|e^{i\theta_k} - y_k\|^2$ is used. Using the complex polar representation of the auxiliary variable $y_k = \rho_k e^{i\phi_k}$, in continuous-time this leads to gradient algorithm

$$\frac{d}{dt}\theta_k = y_k^T q_k = \rho_k \sin(\phi_k - \theta_k) \,, \tag{4.5}$$

where q_k is the vector in $\mathbb{R}^2$ corresponding to $ie^{i\theta_k}$, the direction tangent to the circle at θ_k. In discrete-time it yields

$$\theta_k(t+1) = \arg\left(\beta e^{i\theta_k(t)} + (1-\beta)e^{i\phi_k(t)}\right) \tag{4.6}$$

where $\beta \in (0,1)$ governs the inertia of the update; if $\beta e^{i\theta_k(t)} + (1-\beta)e^{i\phi_k(t)} = 0$, then any point on S^1 can be selected for $\theta_k(t+1)$. The similarity of (4.5) and (4.6) with (3.12) and (3.6) respectively is not surprising since the derivation of the algorithms from the cost function is analogous to Section 3.2.

It is important to check how the introduction of auxiliary variables y_k gets along with the symmetry of the initial configuration space S^1. Algorithms (4.7) and (4.8) are invariant with respect to a uniform translation of the θ_k along S^1 *associated with a corresponding rotation of the* y_k, i.e. changing $(\theta_k,\ y_k)$ to $(\theta_k + a,\ y_k e^{ia})$ for some $a \in \mathbb{R}$ and $\forall k \in \mathcal{V}$. This reflects that the auxiliary variables have a geometrical meaning y_k equivalent to the positions $e^{i\theta_k}$. Algorithms (4.7) and (4.8) satisfy the geometric invariance of the circle, *but independent agents should implement the* y_k *locally with the appropriate geometric meaning.* This is possible by expressing the position of y_k in the plane *relative to* $e^{i\theta_k}$: the actual auxiliary variables used by the agents are $r_k := y_k e^{-i\theta_k}$. With this change of variables, the overall algorithms rewrite (continuous-time)

$$\begin{aligned} \tfrac{d}{dt}\theta_k &= \rho_k \sin(\chi_k) \\ \tfrac{d}{dt}r_k &= \alpha \sum_{j=1}^{N} a_{jk}\left(r_j e^{i(\theta_j-\theta_k)} - r_k\right) - i r_k \tfrac{d}{dt}\theta_k \end{aligned} \tag{4.7}$$

$$\text{with complex polar representation } r_k = \rho_k e^{i\chi_k} \quad , \quad k = 1,2,...,N\ ,$$

and (discrete-time)

$$\begin{aligned} \theta_k(t+1) &= \theta_k(t) + \arg\left(\beta + (1-\beta)e^{i\chi_k(t)}\right) \\ r_k(t+1) &= \left(r_k(t) + \alpha \sum_{j=1}^{N} a_{jk}\left(r_j(t)e^{i(\theta_j(t)-\theta_k(t))} - r_k(t)\right)\right) e^{-i(\theta_k(t+1)-\theta_k(t))} \end{aligned} \tag{4.8}$$

$$\text{with complex polar representation } r_k = \rho_k e^{i\chi_k} \quad , \quad k = 1,2,...,N\ .$$

These are the algorithms actually implemented by the agents. They contain periodic functions of the relative positions of the agents $(\theta_j - \theta_k)$, as well as (arguments or norms of) the auxiliary variables r_k, $k = 1,2,...,N$, which are also *relative* positions of y_k with respect to the position of k on the circle. In particular, the r_k can just be stored and communicated among agents as complex *numbers*, without taking into account that they represent positions in the plane.

Like the y_k, the r_k can be initialized arbitrarily. In order to keep the agents close to their initial positions when they are initially all located within a small subset of S^1, it can be interesting to choose the $y_k(0)$ (or at least their projection on S^1) close to the corresponding $e^{i\theta_k(0)}$. This would mean to choose the $r_k(0)$ close to real positive numbers.

4.3.2 Convergence analysis

Proposition 4.3.1: *Consider a set of N agents interconnected according to a (piecewise continuous) uniformly connected δ-digraph $\mathbb{G}$, applying algorithm (4.7) with $\alpha > 0$, or algorithm*

(4.8) with $\beta \in (0,1)$ and $\alpha \in (0, b/d_k^{(i)})$ $\forall t \geq 0$ and $\forall k \in \mathcal{V}$, for some constant $b \in (0,1)$. Then for almost all initial conditions $(r_k(0), \theta_k(0))$, $k = 1, 2, ..., N$, the solutions converge to an equilibrium where $r_k e^{i\theta_k} = \bar{y} \neq 0$ $\forall k \in \mathcal{V}$ and $\arg(r_k) = m_k \pi$ $\forall k \in \mathcal{V}$, with $m_k \in \{0, 1\}$; if $\mathbb{G}(t)$ is balanced $\forall t \geq 0$, then $\bar{y} = \frac{1}{N} \sum_{k=1}^{N} (r_k(0) e^{i\theta_k(0)})$. Moreover, the set corresponding to synchronization of the θ_k is stable and all solutions outside this set are unstable.

Proof: The evolution of the $y_k = r_k e^{i\theta_k}$ is independent of the θ_k and follows a linear vector space consensus algorithm (3.1),(3.3) or (3.2),(3.3). Thus Propositions 3.1.1 and 3.1.2 imply that the y_k asymptotically synchronize for all initial conditions, and that the consensus value $\bar{y}$ equals $\frac{1}{N} \sum_{k=1}^{N} y_k(0)$ for balanced graphs. For any given graph $\mathbb{G}(t)$, the set of $r_k(0)$ leading to $\bar{y} = 0$ has the dimension of $\mathbb{C}^{N-1}$; it is thus of measure 0 in the space $\mathbb{C}^N$ of all possible $r_k(0)$. Therefore, speaking of "almost all" initial conditions, it can be said that $\bar{y} \neq 0$. Anyway, the set where $\bar{y} = 0$ is unstable with respect to perturbations of the $r_k(0)$.

The rest of the proof considers continuous-time algorithm (4.7); the proof for the discrete-time counterpart follows the same lines. From the previous paragraph, algorithm (4.7) is an asymptotically autonomous system. As recalled in Section 2.3, solutions of an asymptotically autonomous system converge to a chain recurrent set of the limiting system. Here, the limiting system is obtained by replacing r_k with $\bar{y} e^{-i\theta_k}$ in the first equation of (4.7), yielding

$$\frac{d}{dt}\theta_k = \|\bar{y}\| \sin(\arg(\bar{y}) - \theta_k) , \quad k = 1, 2, ..., N . \tag{4.9}$$

Thus the agents are decoupled in the limiting system. For each k, (4.9) is a gradient system for the cost function $V_{\mathrm{proj}\,\bar{y}}(\theta_k) = \frac{\bar{y}}{2} \left(2 \sin(\frac{\arg(\bar{y}) - \theta_k}{2})\right)^2$. From Proposition 2.3.8, the chain recurrent set of a one-dimensional, continuously differentiable gradient system is equal to its equilibrium set. Thus for the overall system (4.7), each θ_k converges to the equilibrium set of (4.9). The latter just contains $\theta_k = \arg(\bar{y})$ and $\theta_k = \arg(\bar{y}) + \pi$. This proves the convergence property. Stability of synchronization follows from the fact that $\theta_k = \arg(\bar{y})$ is a minimum of $V_{\mathrm{proj}\,\bar{y}}$, and thus stable for (4.9); $\theta_k = \arg(\bar{y}) + \pi$ corresponds to a maximum of $V_{\mathrm{proj}\,\bar{y}}$ and the other equilibria are unstable. □

Proposition 4.3.1 indicates that the dynamic controllers (4.7) and (4.8) almost recover the convergence properties of vector space consensus algorithms; this is not surprising since the actual synchronization work is done on auxiliary variables $y_k \in \mathbb{R}^2 \cong \mathbb{C}$. As already mentioned in Section 4.2, the presence of unstable equilibria is unavoidable on a compact configuration space. It can be expected that actually almost all initial conditions converge to the stable equilibrium corresponding to synchronization; indeed, for $y_k = \bar{y} \neq 0$ fixed, an agent applying the first equation of (4.7) and (4.8) can only stay at the unstable equilibrium $\theta_k = \arg(\bar{y}) + \pi$ or asymptotically converge to $\theta_k = \arg(\bar{y})$. It is however not totally clear how to rigorously prove this fact in a time-varying setting with only asymptotically constant $y_k = \bar{y}$.

A last comment can be added concerning the convergence rate. On vector spaces, convergence towards synchronization is exponential from $t = 0$ on for any initial conditions. This is not the case for algorithms (4.7) and (4.8) on the circle. Indeed, convergence is slow when starting close to unstable equilibria of a system; in the present case, this involves not only the configurations where $\theta_k = \arg(\bar{y}) + \pi$ for some k, but also the situation where $\bar{y} = 0$. Thus in practical applications, if it is possible in any way, initial conditions $r_k(0)$ should be chosen to avoid the neighborhood of $\bar{y} = 0$.

Recapitulation

Part I of the dissertation is devoted to the study of synchronization algorithms on the circle, that maintain the geometric symmetry of the circle and the equivalence of all agents in the swarm, with limited interconnections among agents.

It starts by presenting well-known linear "consensus algorithms" (3.1),(3.2),(3.3) that solve this problem on vector spaces. Then it shows that the different topology of the circle requires a modification of the linear consensus algorithm. Corresponding synchronization algorithms on the circle are derived in discrete- and continuous-time (3.6),(3.12), and it is shown how they are directly linked to the Vicsek model and the Kuramoto model previously proposed respectively in [148] and [71] to describe collective phenomena.

Analysis of the synchronization algorithms (3.6),(3.12) shows that their convergence properties are equivalent to those of the linear vector space consensus algorithm when the agents are close enough to each other — that is, all within a semicircle. However, when agents are distributed over the whole circle, the algorithms' behavior is not as clear anymore: for fixed undirected interconnection graph $\mathbb{G}$, the solutions converge to a set of equilibria, but there are equilibria different from synchronization, sometimes even stable ones depending on $\mathbb{G}$; for directed and time-varying $\mathbb{G}$, the situation is more involved and the swarm does not necessarily converge to an equilibrium set. Thus *global* synchronization on S^1 is not so easy anymore, and several questions remain open regarding necessary and sufficient conditions for (3.6),(3.12) to possess particular convergence properties.

Therefore, three new algorithms are proposed with the goal to improve global synchronization properties on the circle. First for fixed undirected $\mathbb{G}$, the interaction profile among agents is modified. As for the synchronization algorithm (3.12), solutions converge to a set of equilibria; however, with the modified profile, synchronization is the *only* asymptotically stable equilibrium, for any connected $\mathbb{G}$. This allows to overcome the presence of spurious stable local equilibria observed in Section 3.4.1. A second alternative proposes to use a Gossip Algorithm to break symmetries that prevent global synchronization: at each time instant, each agent chooses to take into account the position of at most one of its in-neighbors and moves towards it. In order to respect agent equivalence, the in-neighbor choice is random. Global asymptotic synchronization with probability 1 can be proven for uniformly connected $\mathbb{G}$; but only pessimistic, extremely slow bounds can be computed for the "convergence rate". The directed version of the Gossip Algorithm is completely independent of the underlying configuration space and can in fact be applied to reach agreement starting from any set of N initial symbols. A third alternative algorithm overcomes the non-convexity of S^1 by actually synchronizing auxiliary variables in the plane, and letting the agents follow the projection on S^1 of their auxiliary variable. This allows to recover, for almost all initial conditions, the convergence properties of the linear vector space consensus algorithm. However, it requires agents to exchange values of their auxiliary variables, which may not always be possible (from

a design viewpoint) or realistic (from a modeling viewpoint).

An overall conclusion of Part I is that synchronization algorithms on the circle feature several specific phenomena which are absent on vector spaces. This increased richness on S^1 in the behavior of synchronization algorithms motivates Part II. The latter considers the synchronization problem on higher-dimensional "circle-like" manifolds, namely *connected compact homogeneous manifolds*. It exploits the phenomenon of local equilibria to define specific configurations different from synchronization, termed *consensus*, *anti-consensus* and *balancing*; a final chapter also shows how this synchronization framework can be used in a more complex setting with realistic mechanical dynamics.

The original contribution in Chapter 3 mainly consists of unifying different existing approaches for consensus on vector spaces and on the circle; examples in Section 3.4 are also original. The algorithms in Chapter 4 are all original contributions.

The original content of Part I is published in [118, 122, 124].

Part II

Mean and consensus on compact homogeneous manifolds

In the present Part II, the topic of the first two chapters was developed at University of Liège, in collaboration with Dr. Luca Scardovi for issues on the circle, and with occasional help and feedback provided by Prof. Pierre-Antoine Absil. The content of the chapter about rigid body synchronization with a mechanical model was mainly developed during a stay as Visiting Student Research Collaborator in the research unit of Prof. Naomi Leonard at Princeton University.

Introduction The purpose of this second part is to extend the study of position coordination, initiated in Part I, to more general *configurations* — i.e. particular sets of relative positions different from synchronization — and more general *manifolds*. This includes both defining and characterizing particular configurations, as well as designing control laws for individual agents to make a swarm converge towards these configurations.

The main ingredients of Part I are maintained: limited interconnection among agents, invariance with respect to uniform translations of all agents and related use only of *relative* positions in the controllers, and configuration spaces whose global topology fundamentally differs from vector spaces. Explicitly, the configuration spaces considered are *connected compact homogeneous manifolds*. A formal definition is provided in the mathematical background, Section 2.4, and recalled at the beginning of Chapter 5. Informally, a compact homogeneous manifold can be thought of as a compact space which is so highly symmetric that "all points are equivalent"; examples of connected compact homogeneous manifolds include the n-dimensional sphere S^n, compact Lie groups like the group of n-dimensional rotation matrices $SO(n)$, and the Grassmann manifolds $Grass(p, n)$ of any dimensions. From a theoretical point of view, focusing on manifolds on which "all points are equivalent" is sensible for a dissertation whose main concern is invariance and symmetry with respect to absolute ($\neq$ relative) positions; connectedness is just necessary to ensure that agents starting at arbitrary initial conditions can always be driven towards a common point by continuous trajectories; compactness clearly differentiates the objects under study from vector spaces and ensures that trajectories are bounded. Although most of the fundamental difficulties already exist on the circle, Part II provides a more general geometric view on them. It thereby formalizes and tackles more general problems than synchronization, namely the definition of a computationally simple, globally defined "mean position" on manifolds and the definition and stabilization of "distributed" configurations called *consensus*, *anti-consensus* and *balanced*. "Maximally distributed configurations", as balanced configurations could be called, only make sense on compact configuration spaces: on a vector space it would just amount to driving the agents to infinity in different directions. Another ingredient of Part II is to show how the proposed framework, including Parts I and III, can be incorporated into a dynamically more realistic framework of mechanical systems. Indeed, Part I, Part III and most of Part II only consider first-order, velocity-controlled dynamics. It is thus important to show that this is not a fundamental limitation of the present dissertation; to that end, Chapter 7 considers the particular example of synchronization on $SO(3)$ with torque-controlled, mechanical rigid body dynamics.

In engineering applications, driving agents both towards each other or away from each other are ubiquitous basic tasks of growing interest. The present study is directly linked to the vast literature about the circle, involving engineering applications and the study of basic synchronization mechanisms (see the Introduction to Part I for many references on this topic). The general theory is motivated by the fact that many applications involve manifolds

that are not isomorphic to a vector space, but fit into the framework of connected compact homogeneous manifolds.

The task of "moving towards each other" is linked to the distributed computation of means/averages of datasets in algorithmic settings, like distributed decision making (e.g. [95, 104, 145]), neural and communication networks (e.g. [51, 144]) and clustering or other reduction methods (e.g. [45]), as well as to synchronization control of physical agents for instance for autonomous swarm/formation operation (e.g. [62, 70, 76, 131]). For vehicle formations moving in $\mathbb{R}^2$ or $\mathbb{R}^3$ for instance, the agents' orientations evolve in manifolds $SO(2) \cong S^1$ or $SO(3)$; this is also important in Part III. $SO(3)$ is useful in more general settings for orientation control of 3-dimensional bodies, like underwater or aerial vehicles (e.g. [74]). By far the most popular field of interest for synchronization on $SO(3)$ is spacecraft attitude control (see references in the following literature review). This is probably due on one hand to the modeling fact that invariance with respect to absolute orientation is a good approximation in the microgravity environment and on the other hand to the current study of space missions for the near future which require accurate attitude synchronization, for instance for in-orbit assembly (e.g. [55, 86]) or space interferometry (e.g. the Darwin mission, see [9]).

The task of "moving away from each other" implies spreading the agents in the available configuration space, which might be useful in algorithmic settings e.g. for optimal covering or coding (e.g. [7, 8, 28]), or for physical exploration purposes (e.g. [29, 39]). For instance, using a swarm of sensor platforms for global planet exploration would involve "optimal spreading" on the sphere S^2. Practical applications of optimal distributions on the Grassmann manifolds rather appear in algorithmic problems; [28] mentions the optimal placement of N laser beams for cancer treatment and the projection of multi-dimensional data onto N representative planes. As for Part I, more on potential applications can be found in the Introduction chapter of the thesis.

Outline and Main points In accordance with the central concern of symmetries, the focus in Part II is on connected compact *homogeneous manifolds*, on which "each point is equivalent". Symmetry with respect to any uniform tranlation of all agents on the manifold is imposed and allows to focus on *configurations* of the swarm, i.e. specific *relative positions* of the agents. A central requirement for the *control algorithms* is that they must maintain the invariance of the configuration space, which implies that they may only depend on *relative* positions as well. This leads to a fundamental agreement problem on connected compact homogeneous manifolds. Agent models are mostly considered in continuous-time. Chapters 5 and 6 consider velocity-controlled simple integrators like in Part I, but the purpose of Chapter 7 is precisely to discuss how this adapts to more general dynamics.

Mathematical background about embedded homogeneous manifolds, and more specifically the manifolds $SO(n)$ and $Grass(p, n)$, can be found in Section 2.4.

As shown in Section 3.1, synchronization algorithms are well understood in Euclidean spaces. They are in fact based on the natural definition and distributed computation of the *centroid*, or mean position, in $\mathbb{R}^m$. Therefore Chapter 5 starts in Section 5.1 by defining and studying the *induced arithmetic mean*, an alternative mean position on manifolds that may be of independent interest. The main idea is to embed the manifold $\mathcal{M}$ in $\mathbb{R}^m$ and measure distances *in* $\mathbb{R}^m$ between agents; this is equivalent to computing a centroid in the embedding space $\mathbb{R}^m$ of $\mathcal{M}$ and projecting it onto $\mathcal{M}$. Although the induced arithmetic mean differs from

the traditional Karcher mean, it has a clear geometric meaning with the advantage of being easily computable — analytical solutions are provided for $SO(n)$ and $Grass(p, n)$. Building on the induced arithmetic mean, Section 5.2 introduces and discusses related definitions of *consensus* configurations, which are more general than synchronization, and *anti-consensus* and *balanced* configurations, which correspond to "repulsion" of the agents on the manifold and can hence be seen as "maximally different" from synchronization. Section 5.3 formulates these diverse configurations in terms of extrema of a cost function, in order to pave the way for algorithm design.

Chapter 6 considers the design of control algorithms for individual velocity-controlled agents such that the swarm converges to synchronization, consensus, anti-consensus or balanced configurations. The algorithms satisfy proper invariance properties and take limited interconnections among agents into account, as in Part I; in fact, the proposed algorithms generalize those of Part I. Section 6.1 derives a gradient algorithm generalizing the one of Section 3.2 and examines its convergence properties: (anti-)consensus configurations are its only stable equilibria for fixed undirected graphs. Section 6.2 examines the generalization of the algorithms proposed in Chapter 4 for the circle. Section 6.2.1 considers a dynamic controller with auxiliary variables, proposed for global synchronization on the circle in Section 4.3. It derives extensions on connected compact homogeneous manifolds to achieve (almost) global convergence to synchronization and to the anti-consensus configurations of the complete graph (in practice, this seems to imply that the swarm converges to a balanced configuration), for uniformly connected communication graphs. These controllers employ an auxiliary variable that evolves in the embedding space $\mathbb{R}^m$; communication of these variables between agents is briefly discussed. Finally, mainly for the sake of completeness, Sections 6.2.2 and 6.2.3 consider the extension of the algorithms proposed for the circle in Sections 4.1 and 4.2 respectively, i.e. modifying the interaction profile and introducing random link selection.

Throughout Chapters 5 and 6, concepts are illustrated on the special orthogonal group $SO(n)$, the Grassmann manifold $Grass(p, n)$ of p-dimensional vector spaces in $\mathbb{R}^n$, and sometimes the circle S^1, which is in fact isomorphic to both $SO(2)$ and $Grass(1, 2)$. Other connected compact homogeneous manifolds to which this framework could be applied include the n-dimensional sphere S^n and the connected subgroups of $SO(n)$.

Chapter 7 is mainly conceived as an explicit illustration of how the geometrical concepts and algorithms developed for single-integrator agents can be applied in settings with more complete mechanical agent dynamics. For this purpose, it considers the example of rigid body rotation in 3 dimensions, involving (i) on the manifold side, the compact Lie group $SO(3)$ and (ii) on the system dynamics side, the torque-controlled Euler equations for rotational motion of isolated rigid bodies in Newtonian mechanics. Although a more general setting could be considered at that place (e.g. in line with [5]), it is chosen to focus on a concrete example for the benefit of easier understanding. As mentioned above, the chosen example is also relevant in practice, among others for satellite attitude control applications.

The chapter starts with a few remarks, among others about the possibility to use the popular unit quaternion parametrization of $SO(3)$ in the coordination context. Then Section 7.1 briefly reviews the main reasons why synchronization of mechanical bodies differs from synchronization of first-order integrators. First the task of physical position synchronization, involving points on a vector space, is briefly considered. Then it is shown that physical orientation synchronization, involving the manifold $SO(3)$, introduces an additional difficulty because the mechanical dynamics are more complex: unlike Newton's equation for linear

motion of points, the mechanical equation for free rigid body rotations contains a nonlinear function of the velocity. Sections 7.2 and 7.3 propose and analyze control algorithms for "attitude synchronization" — the task of controlling towards the same orientation a swarm of rigid bodies evolving according to the laws of Newtonian mechanics. Control laws for attitude synchronization abound in the literature (see the next paragraph), but for most of them all the agents follow a reference or leader, which is not compatible with the requirement of invariance with respect to absolute orientation and circumvents the associated agreement problem. Section 7.2 essentially proposes to replace the tracking of a reference by a "consensus tracker", tracking the angular velocity which a first-order integrator synchronization algorithm would impose. Section 7.3 explores another technique known as *energy shaping* to obtain attitude synchronization. It first shows that an existing result in [96, 98, 109, 135], achieving attitude synchronization with algorithms that are indeed invariant with respect to absolute orientation, is actually directly linked to the developments in the present part of the dissertation: the cost function used to derive gradient algorithms in Chapter 6 is used in the mechanical setting as an artificial potential to derive control torques. Sections 7.3.2 and 7.3.3 then propose two separate extensions of the existing algorithms.

1. Control torques in [96, 98, 109, 135] use angular velocities in a way that drives the swarm to synchronization at rest (or another a priori specified motion). Section 7.3.2 proposes a control torque depending on *relative velocities* only, for which the swarm behaves more like a single rigid body: any synchronized free rigid body rotation is a solution for the controlled swarm.
2. The existing results are valid for fixed, undirected communication graphs. Section 7.3.3 adapts the "auxiliary variable" strategy of Section 6.2.1 to the energy shaping method, in order to obtain almost global attitude synchronization with directed and time-varying communication graphs.

Related literature Most of the work related to synchronization and balancing on manifolds concerns the circle S^1. Relevant literature is listed in Part I. To this can be added that applications of collective planar motion control in [131, 132] not only use synchronization algorithms on S^1, but also various forms of balancing algorithms for distribution of the agents on circular trajectories. The notion of "balanced configurations" was first introduced in [131] for the specific case of the circle. The use of auxiliary variables to enhance convergence properties towards synchronization and balancing has been applied in [123, 126, 130, 132] for single integrator dynamics on manifolds. Dynamic controllers with communicated auxiliary variables also allow to obtain global synchronization of linear systems on vector spaces, as shown in [125]; with static controllers, synchronization of e.g. double integrators based on relative velocity and position measurements only requires additional conditions on the interconnection graph and on controller gains, see [112].

The popular, application-driven subject of attitude synchronization has attracted attention towards the manifold $SO(3)$; several authors present algorithms that asymptotically synchronize satellite attitudes with torque-controlled, mechanical dynamics. They often rely on tracking a common external reference (e.g. [72, 86, 89, 110, 147]) or leader (e.g. [15, 70]), which introduces elements that the present dissertation seeks to avoid, see Section 1.1. One paper proposing algorithms for attitude synchronization without reference or leader track-

ing is [109]. All these results focus on nonlinear controller design, and most of them use the practically convenient and popular quaternion parametrization for $SO(3)$. However the unitary quaternion parametrization is not a *representation* of $SO(3)$ in the mathematical sense: it contains two elements for each point of $SO(3)$. It is therefore dangerous to design quaternion-based algorithms when considering $SO(3)$ more than locally, because differently treating two quaternions that represent the same point of $SO(3)$ can quickly introduce unwanted artefacts and break the global geometry and invariance of the problem. The research group of Prof. N. Leonard has taken a more geometric approach to attitude synchronization, see [48, 96, 97, 98, 99, 100, 135], with energy shaping. Section 7.3 directly extends these results, as previously explained. It turns out that the distance measure proposed for consensus in Chapter 5 coincides on $SO(3)$ with the distances proposed in [96, 135] and in [22].

More generally, the geometric viewpoint for control design of mechanical systems has become a fundamental subject in control theory; see for instance tools and results in [10, 12, 21, 22, 83, 133, 146].

The basic problem to define and compute a mean or centroid of points on a manifold $\mathcal{M}$ has attracted some attention, as can be seen from [23, 41, 52] among others. The "projected arithmetic mean" defined in [92] for $SO(3)$ coincides with the mean proposed in Section 5.1. In fact, the basic idea of computing statistics in a larger and simpler embedding manifold (usually Euclidean space) and projecting the result back onto the original manifold, goes back to 1972, see [37].

A short example in [1] addresses the computation of a "centroid of subspaces", i.e. a centroid on $Grass(p, n)$, without much theoretical analysis; in fact, the algorithms in [1] are similar to those of Section 6.1 on $Grass(p, n)$. More recently, [45] uses the centroid associated to the projector representation of $Grass(p, n)$, exactly as defined in Section 5.1 but without going into theoretical details, to compute the cluster centers in a clustering algorithm. The distance measure associated (also in Section 5.1) to this centroid on $Grass(p, n)$ is called the *chordal distance* in [8, 28], where it is used to derive optimal distributions ("packings") of N agents on some specific Grassmann manifolds.

More generally, the topic of optimization-based algorithm design on manifolds has considerably developed over the last decade; see e.g. [18, 38] and the books [2, 49].

Chapter 5

Defining consensus configurations on manifolds

Consider a set of N agents on a vector space $\mathbb{R}^m$, with interconnections among agents characterized by an unweighted graph. The position of agent k is denoted by x_k, $\forall k \in \mathcal{V} = \{1, 2, ..., N\}$. The *centroid*, or *arithmetic mean position*, of the set of agents is defined as

$$C_e(\{x_k : k \in \mathcal{V}\}) = \frac{1}{N} \sum_{k=1}^{N} x_k \, . \tag{5.1}$$

With this definition of centroid, the basic control law (3.3) can be rewritten

$$u_k(t) = \alpha \, d_k^{(i)} \, (\, C_e(\{x_j(t) : j \rightsquigarrow k\}) - x_k(t) \,) \, , \quad k = 1, 2..., N \, .$$

This shows that the linear vector space consensus algorithms actually correspond to agent k moving towards the centroid of its in-neighbors. The present part of the dissertation uses this characterization to build consensus algorithms on connected compact homogeneous manifolds. Therefore this chapter studies a computationally simple definition of mean position on such manifolds and links it to a consensus framework. The next chapter derives the related algorithms.

A homogeneous manifold $\mathcal{M}$ is a manifold isomorphic to the quotient manifold $\mathcal{G}/\mathcal{H}$ of a Lie group $\mathcal{G}$ by one of its subgroups $\mathcal{H}$, see Section 2.4. Informally, it can be seen as a manifold on which "all points are equivalent". The present dissertation considers connected compact homogeneous manifolds satisfying the following embedding property.

Assumption 5.0.2: *$\mathcal{M}$ is a connected compact homogeneous manifold $\mathcal{G}/\mathcal{H}$ smoothly embedded in $\mathbb{R}^m$ with the Euclidean norm $\|x\| = r_{\mathcal{M}}$ constant over $x \in \mathcal{M}$. The Lie group $\mathcal{G}$ acts as a subgroup of the orthogonal group on $\mathbb{R}^m$.*

It is a well-known fact of differential geometry that any smooth $\frac{m}{2}$-dimensional Riemannian manifold can be smoothly embedded in $\mathbb{R}^m$. The additional condition $\|x\| = r_{\mathcal{M}}$ is in agreement with the fact that all points on $\mathcal{M}$ should be equivalent. It is sometimes preferred to represent $x \in \mathcal{M}$ by a matrix $B \in \mathbb{R}^{n_1 \times n_2}$ instead of a vector; this is particularly useful for matrix Lie groups, which are naturally represented by square matrices. Componentwise identification $\mathbb{R}^{n_1 \times n_2} \cong \mathbb{R}^m$ is assumed whenever necessary; the corresponding norm is the

Frobenius norm $\|B\| = \sqrt{\text{trace}(B^T B)}$.

A central concern is that the natural invariance of the setting must be maintained, in definitions as well as in control interact. Therefore, all the definitions and algorithms in Part II must be invariant with respect to applying a common transformation of the symmetry group $\mathcal{G}$ to the positions of all the agents; that means, the swarm and its characterizations must behave exactly in the same way after any transformation $(x_1, x_2, ..., x_N) \to (g(x_1), g(x_2), ..., g(x_N))$, where $g \in \mathcal{G} : \mathcal{M} \to \mathcal{M}$ is a transformation belonging to the symmetry group of the homogeneous manifold. A *configuration* must be understood in the sense of Definition 3.4.1 where a "uniform translation" of all the agents is defined as such a symmetry transformation.

5.1 The induced arithmetic mean

Consider a set of N agents on a manifold $\mathcal{M}$ satisfying Assumption 5.0.2. The position of agent k is denoted by x_k and a weight $w_k \in \mathbb{R}_{>0}$ reflects its "importance" in the swarm, $\forall k \in \mathcal{V} = \{1, 2, ..., N\}$. The weights w_k are only introduced in this first section for the sake of completeness, in analogy with the masses m_k of individual particles when computing their center of mass; diversified motivations can be behind the w_k, but to respect geometric invariance they may not depend on the position of agent k on $\mathcal{M}$. Further sections will assume $w_k = 1\ \forall k \in \mathcal{V}$ in order to retrieve a strict equivalence of all agents.

Definition 5.1.1: *The **induced arithmetic mean** IAM $\subseteq \mathcal{M}$ of N agents of weights $w_k > 0$ and positions $x_k \in \mathcal{M}$, $k = 1, 2, ..., N$, is the set of points on $\mathcal{M}$ that globally minimize the weighted sum of squared Euclidean distances in $\mathbb{R}^m$ to each x_k:*

$$IAM \ = \ \underset{c\in\mathcal{M}}{\text{argmin}} \sum_{k=1}^{N} w_k \, d^2_{\mathbb{R}^m}(x_k, c) \ = \ \underset{c\in\mathcal{M}}{\text{argmin}} \sum_{k=1}^{N} w_k \, (x_k - c)^T (x_k - c) \, . \tag{5.2}$$

*The **anti-[induced arithmetic mean]** AIAM $\subseteq \mathcal{M}$ is the set of points on $\mathcal{M}$ that globally maximize the weighted sum of squared Euclidean distances* in $\mathbb{R}^m$ *to each x_k:*

$$AIAM \ = \ \underset{c\in\mathcal{M}}{\text{argmax}} \sum_{k=1}^{N} w_k \, d^2_{\mathbb{R}^m}(x_k, c) \ = \ \underset{c\in\mathcal{M}}{\text{argmax}} \sum_{k=1}^{N} w_k \, (x_k - c)^T (x_k - c) \, . \tag{5.3}$$

The terminology is derived from [92] where the *IAM* on $SO(3)$ is called the *projected arithmetic mean*. The main point in Definition 5.1.1 is that distances are measured *in the embedding space* $\mathbb{R}^m$. It thereby differs from the canonical definition of mean of N agents on $\mathcal{M}$, the *Karcher mean* [44, 52, 65, 108]. The latter uses instead the *geodesic distance* $d_{\mathcal{M}}$ along $\mathcal{M}$ with the Riemannian metric induced by the embedding of $\mathcal{M}$ in $\mathbb{R}^m$:

$$C_{Karcher} = \underset{c\in\mathcal{M}}{\text{argmin}} \sum_{k=1}^{N} w_k \, d^2_{\mathcal{M}}(x_k, c) \, .$$

The induced arithmetic mean has the following properties.

- The *IAM* of a single point x_1 is the point itself.
- The *IAM* is invariant under permutations of agents of equal weight.
- The *IAM* commutes with the symmetry group of the homogeneous manifold.

- The *IAM* does not always reduce to a single point.

The last feature seems unavoidable for any mean (including the Karcher mean) that satisfies the other properties. The main advantage of the *IAM* over the Karcher mean is computational. The distance among points on $\mathcal{M}$ defined as the Euclidean distance in the embedding space $\mathbb{R}^m$ is called the *chordal distance* in several situations, see examples below; for the circle $\mathcal{M} = S^1$, it is the length of the *chord* joining the points in $\mathbb{R}^2$. Unlike the geodesic distance, the squared chordal distance is smooth everywhere. However, it does not derive from a metric. On the circle $\mathcal{M} = S^1$, the geodesic distance derived from the canonical metric associated to its embedding in $\mathbb{R}^2$ is the arclength distance, which is not differentiable for an arclength of π. When all the points are in an infinitesimal subset of $\mathcal{M}$, the relative error (i.e. relative to the maximal distance between agents) between geodesic and chordal distances is expected to decrease to zero, such that the Karcher mean and *IAM* can be expected to become close; a quantitative statement of this assertion is derived for the circle in Example 5.1.3.

On vector spaces, the *IAM* and Karcher mean are identical and correspond to the *centroid* or *arithmetic mean* $C_e = \frac{1}{W} \sum_{k=1}^{N} w_k \, x_k$, with $W = \sum_{k=1}^{N} w_k$. On compact homogeneous manifolds $\mathcal{M}$, the *IAM* and *AIAM* are closely related to the centroid $C_e = \frac{1}{W} \sum_{k=1}^{N} w_k \, x_k$ of $x_k \in \mathcal{M} \subset \mathbb{R}^m$. Since $\|c\| = r_{\mathcal{M}}$ for $c \in \mathcal{M}$ by Assumption 5.0.2, equivalent definitions for the *IAM* and *AIAM* are

$$IAM = \underset{c \in \mathcal{M}}{\operatorname{argmax}}(c^T \, C_e) \qquad \text{and} \qquad AIAM = \underset{c \in \mathcal{M}}{\operatorname{argmax}}(-c^T \, C_e) \; . \tag{5.4}$$

Hence, computing the *IAM* and *AIAM* just involves a search for the global maximizers of a linear function in a regular search space $\mathcal{M} \subset \mathbb{R}^m$. Local maximization methods even suffice if the linear function has no maxima on $\mathcal{M}$ other than the global maxima. This is the case for any linear function on $SO(n)$ and on $Grass(p, n)$ (see further) as well as on the n-dimensional sphere S^n in $\mathbb{R}^{n+1}$. Not knowing whether it holds for all manifolds satisfying Assumption 5.0.2, the following blanket assumption is formulated.

Assumption 5.1.2: *The local maxima of any linear function $f(c) = c^T \, b$ over $c \in \mathcal{M}$, with b fixed in $\mathbb{R}^m$, are all global maxima.*

The following examples exclusively consider the *IAM*; from (5.4), the conclusions for the *AIAM* are obtained by replacing C_e with $-C_e$.

Ex. 5.1.3: ***IAM* on the circle:** The circle embedded in $\mathbb{R}^2$ with its center at the origin satisfies Assumptions 5.0.2 and 5.1.2. The *IAM* is simply the central projection of C_e onto the circle, see Figure 5.1. Hence it corresponds to the whole circle if $C_e = 0$ and else reduces to a single point. The *IAM* uses the chordal distance between points, while the Karcher mean would use arclength distance; see Figure 5.2 in Section 5.3.

Consider a set of N points that are in a small subset of the circle, without loss of generality in a small interval $(-\varepsilon, \varepsilon)$ around 0. The arclength distance between points θ_k and θ_j is simply $d_{S^1}(\theta_k, \theta_j) = |\theta_j - \theta_k|$, whereas the chordal distance is $d_{\mathbb{R}^2}(\theta_k, \theta_j) = 2|\sin(\frac{\theta_j - \theta_k}{2})|$; the difference on the distance is thus of order $\mathcal{O}(\theta_k - \theta_j)^3$, that is of relative order $\mathcal{O}(\theta_k - \theta_j)^2$ with respect to the actual distance. Computing the Karcher mean of several points leads to the condition

$$\sum_{k=1}^{N} (C_{Karcher} - \theta_k) = 0 \quad \Leftrightarrow \quad C_{Karcher} = \tfrac{1}{N} \sum_{k=1}^{N} \theta_k \; .$$

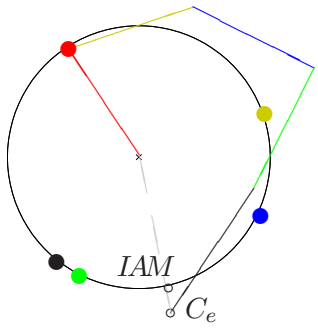


Figure 5.1: Induced arithmetic mean on the circle

Computing the induced arithmetic mean leads to the condition

$$\sum_{k=1}^{N} \sin(IAM - \theta_k) = 0 \quad \Leftrightarrow \quad IAM \simeq \tfrac{1}{N} \sum_{k=1}^{N} \theta_k + \frac{(IAM - \theta_k)^3}{6} \,. \tag{5.5}$$

The *IAM*, like $C_{Karcher}$, belongs to the interval $(-\varepsilon, \varepsilon)$. Therefore, the last term in (5.5) is at most of order $\mathcal{O}(\varepsilon^3)$. Thus the difference between the Karcher mean and the *IAM* is at most of order $\mathcal{O}(\varepsilon^3)$ for agents in an infinitesimal set of size ε, implying a relative error (with respect to the size of the set containing the agents) of order $\mathcal{O}(\varepsilon^2)$.

The squared norm of the centroid $\|C_e\|^2$ for the circle is strictly equal to the "complex order parameter" used for the study of synchronization in the context of the Kuramoto model, see Section 3.2 in Part I of this dissertation. ◇

Ex. 5.1.4: ***IAM* on the special orthogonal group:** The embedding of $SO(n)$ as orthogonal matrices $Q \in \mathbb{R}^{n\times n}$, $\det(Q) > 0$, satisfies Assumption 5.0.2 with $\|Q\| = \sqrt{\text{trace}(Q^T Q)} = \sqrt{n}$. It also satisfies Assumption 5.1.2 (proof in Section 6.1). $C_e = \frac{1}{N}\sum_k Q_k$ is a general $n \times n$ matrix. The *IAM* is linked to the polar decomposition (see Section 2.1) of C_e. Indeed, orthogonal matrix U from the polar decomposition $B = UR$ of $B \in \mathbb{R}^{n\times n}$ is a global minimizer of $d_{\mathbb{R}^{n\times n}}(c, B)$ over $c \in O(n)$. Thus, if $\det(C_e) \geq 0$, then the *IAM* contains all matrices U: $\det(U) > 0$ obtained from the polar decomposition of C_e; this was already noticed in [92]. When $\det(C_e) < 0$, the result is more complicated but still has a closed-form solution.

Proposition 5.1.5: *Consider the polar decomposition $C_e = UR$. The IAM of N points on $SO(n)$ is characterized as follows.*

- *If $\det(C_e) \geq 0$, then $IAM = \{U : \det(U) > 0\}$. It reduces to a single point if the multiplicity of 0 as an eigenvalue of C_e is less than or equal to 1.*
- *If $\det(C_e) \leq 0$, then $IAM = \{UHJH^T\}$ where $\det(U) < 0$, H contains the orthonormalized eigenvectors of R with an eigenvector corresponding to the smallest eigenvalue in the first column, and $J = \begin{pmatrix} -1 & 0 \\ 0 & I_{n-1} \end{pmatrix}$. The IAM reduces to a single point if the smallest eigenvalue of R has multiplicity 1.*

Proof: The proof is provided in Section 6.1 after introducing further necessary material. The critical points of $c^T C_e$ over $c \in SO(n)$ are computed and the local maxima are selected among

them; this is sufficient since $SO(n)$ satisfies Assumption 5.1.2. □

◇

Ex. 5.1.6: *IAM* **on the Grassmann manifold:** See Section 2.4 for more information about the Grassmann manifolds and their representations. The representation of $Grass(p,n)$ with p-bases X_k is not an embedding and cannot be used in the proposed framework; the p-dimensional subspace of $\mathbb{R}^n$ spanned by the columns of $C_e = \frac{1}{N}\sum_k X_k$ would depend on the particular matrices X_k chosen to represent the subspaces $\mathcal{X}_k$. However, the *IAM* can be properly defined with the projector representation, which is an embedding of $Grass(p,n)$ in $\mathbb{S}_n^+$ (to which is attributed the Euclidean metric associated with the Frobenius norm). This "projector embedding" satisfies Assumption 5.0.2; the Frobenius norm of a p-rank projector is $\sqrt{p}$. It also satisfies Assumption 5.1.2 (proof in Section 6.1). The centroid C_e of N projectors is a symmetric positive semidefinite matrix of rank $\geq p$.

Proposition 5.1.7: *The IAM contains all dominant p-eigenspaces of C_e. It reduces to a single point if the p-largest and $(p+1)$-largest eigenvalues of C_e are different.*

Proof: The proof follows the same lines as for $SO(n)$; it is similarly postponed to Section 6.1 because further material is necessary. □

In fact, for $\mathcal{X} \in Grass(p,n)$ with a p-basis representation X and the projector $\Pi_{\mathcal{X}} = XX^T$, the cost function in (5.4) becomes

$$f(\Pi_{\mathcal{X}}) = \text{trace}(\Pi_{\mathcal{X}} C_e) = \text{trace}(X^T C_e X) = \text{trace}(X^T C_e X\,(X^T X)^{-1}) \tag{5.6}$$

where the last expression is equal to the generalized Rayleigh quotient for the computation of the dominant p-eigenspace of C_e. The computation of eigenspaces from cost function (5.6) is extensively covered in [1, 2]. Furthermore, for Π_k being the projector representation of $\mathcal{X}_k \in Grass(p,n)$, it is a well-known fact of linear algebra that the p largest eigenvalues (the others being 0) of $\Pi_{\mathcal{X}}\Pi_k$ are the squared cosines of the principal angles ϕ_k^i, $i = 1, 2, ..., p$, between subspaces $\mathcal{X}$ and $\mathcal{X}_k$. This provides a geometrical interpretation for the *IAM* of subspaces: it minimizes the sum of squared sines of principal angles between the set of subspaces $\mathcal{X}_k$, $k \in \mathcal{V}$, and a centroid candidate subspace $\mathcal{X}$, i.e.

$$IAM = \underset{\mathcal{X}}{\text{argmin}} \sum_{k=1}^{N} \sum_{i=1}^{p} \sin^2(\phi_k^i)\ .$$

The Karcher mean admits the same formula with $\sin^2(\phi_k^i)$ replaced by $(\phi_k^i)^2$ [28]. ◇

5.2 Consensus

The rest of the chapter assumes equal weights $w_k = 1\ \forall k \in \mathcal{V}$; extension to weighted agents is straightforward. Suppose that the N agents evolving on $\mathcal{M}$ are interconnected according to a fixed digraph $\mathbb{G}$ of adjacency matrix A whose elements are denoted a_{jk}. The following definitions are introduced.

Definition 5.2.1: ***Synchronization*** *is the configuration where* $x_j = x_k \quad \forall j,k \in \mathcal{V}$.

Definition 5.2.2: *A **consensus** configuration with graph $\mathbb{G}$ is a configuration where each agent k is located at a point of the IAM of its neighbors $j \rightsquigarrow k$, weighted according to the strength of the corresponding edge:*

$$x_k \in \underset{c\in\mathcal{M}}{\operatorname{argmax}} \left(c^T \sum_{j=1}^{N} a_{jk}\, x_j \right) \quad , \qquad \forall k \in \mathcal{V} \, . \tag{5.7}$$

Definition 5.2.3: *An **anti-consensus** configuration with graph $\mathbb{G}$ is a configuration where each agent k is located at a point of the AIAM of its neighbors $j \rightsquigarrow k$, weighted according to the strength of the corresponding edge:*

$$x_k \in \underset{c\in\mathcal{M}}{\operatorname{argmin}} \left(c^T \sum_{j=1}^{N} a_{jk}\, x_j \right) \quad , \qquad \forall k \in \mathcal{V} \, . \tag{5.8}$$

Thus consensus is defined as a Nash equilibrium: each agent minimizes its own cost function *assuming the others fixed*; the possibility to decrease cost functions by moving several agents simultaneously is not considered. This definition of consensus is motivated by taking an agent-centered viewpoint: an agent looking at its fellows will consider that "it has done its best" if it is located at the *IAM* of their positions. If all agents "have done their best", it is natural to speak of a consensus situation. On vector spaces, for any weakly connected $\mathbb{G}$, the only possible consensus configuration is synchronization; therefore there is no conflict with the meaning given to "consensus on vector spaces". On compact connected homogeneous manifolds, consensus configurations are graph-dependent. Nevertheless, the following link exists between consensus and synchronization.

Proposition 5.2.4: *If $\mathbb{G}$ is an equally-weighted complete graph, then the only possible consensus configuration is synchronization.*

Proof: At consensus the x_k satisfy $x_k^T \sum_{j\neq k} x_j \geq c^T \sum_{j\neq k} x_j \quad \forall c \in \mathcal{M}$, and $\forall k \in \mathcal{V}$. Furthermore, $x_k^T x_k > c^T x_k$ for any $c \in \mathcal{M} \setminus \{x_k\}$. As a consequence, $x_k^T \sum_{j=1}^{N} x_j > c^T \sum_{j=1}^{N} x_j \quad \forall c \in \mathcal{M} \setminus \{x_k\}$ and $\forall k \in \mathcal{V}$. Thus according to (5.4), each x_k is not only located at the *IAM* of all the agents except itself, but also at the *IAM* of *all* the agents, including itself; moreover, the latter reduces to a single point. Thus $x_k = x_j = IAM(\{x_l : l \in \mathcal{V}\})\ \forall k, j$. □

Depending on $\mathbb{G}$, computing all possible consensus or anti-consensus configurations can be difficult, like for Nash equilibria. Sometimes it can be easy, as in the previous Proposition; it has currently not been clearly identified which situations are easy or difficult to handle.

O.Q.: It is currently unknown for which graphs all possible consensus and/or anti-consensus configurations can be easily characterized, even on a particular manifold like S^1. In particular, it is not known for which graphs there exist no other consensus configurations than synchronization.

Synchronization is a configuration of complete consensus. The following is a meaningful way to similarly characterize a configuration of complete anti-consensus.

Definition 5.2.5: *N agents are in a **balanced** configuration if their IAM contains all $\mathcal{M}$.*

Balancing implies some spreading of the agents on the manifold. As for consensus and anti-consensus, a full characterization of balanced configurations seems complicated. Balanced configurations do not always exist (typically, when the number of agents is too small) and are mostly not unique (they can appear in qualitatively different forms). The following link exists between anti-consensus for the equally-weighted complete graph and balancing.

Proposition 5.2.6: *All balanced configurations are anti-consensus configurations for the equally-weighted complete graph.*

Proof: For the equally-weighted complete graph, (5.8) can be written

$$x_k \in \underset{c\in\mathcal{M}}{\operatorname{argmin}} \left(c^T \left(N\, C_e - x_k\right)\right) \qquad \forall k \in \mathcal{V}\ . \tag{5.9}$$

Assume that the agents are balanced. This means that $f(c) = c^T\, C_e$ must be constant over $c \in \mathcal{M}$. Therefore (5.9) reduces to $x_k = x_k\ \forall k$ which is trivially satisfied. □

In contrast to Proposition 5.2.4, Proposition 5.2.6 does not establish an equivalence relation. And indeed, anti-consensus configurations for the equally-weighted complete graph that are not balanced do exist, although they seem to be exceptional.

In fact, for large enough N, there exists a continuum of balanced configurations. Indeed, consider N_{min} the minimum number of agents necessary to build balanced configurations and assume that $N \geq 2N_{min}$. Split the agent set $\mathcal{V}$ into $\mathcal{V}_1$ and $\mathcal{V}_2$ with $|\mathcal{V}_1| > N_{min}$ and $|\mathcal{V}_2| > N_{min}$, and build balanced configurations for $\mathcal{V}_1$ and $\mathcal{V}_2$ separately; these configurations can be arbitrarily translated on $\mathcal{M}$ by the symmetry group. It is not difficult to verify that the overall configuration for $\mathcal{V}$ is then balanced, whatever the translations applied to the configurations for $\mathcal{V}_1$ and $\mathcal{V}_2$; keeping e.g. $\mathcal{V}_1$ fixed and scanning all possible translations of $\mathcal{V}_2$ yields a continuum of different balanced configurations. This is just an example and other continua which are not based on splitting $\mathcal{V}$ can exist as well. From Proposition 5.2.6, this also means that enumerating all possible anti-consensus configurations for the complete graph is not so simple, in contrast to consensus configurations (Proposition 5.2.4).

Ex. 5.2.7: consensus and balancing on the circle: On S^1, a balanced configuration means that the centroid C_e of the agents is located at the centre of the circle. Anti-consensus configurations for the equally-weighted complete graph are characterized in [131]: the only anti-consensus configurations that are not balanced correspond to $(N+1)/2$ agents at one position and $(N-1)/2$ agents at the opposite position on the circle, for N odd. Balanced configurations are unique for $N = 2$ and $N = 3$ and form a continuum for $N > 3$.

The local minima of V_θ in Example 3.4.2 for the undirected ring graph, where consecutive agents are separated by a constant angle $\chi < \pi/2$, are consensus configurations. The local maxima of V_θ where consecutive agents are separated by a constant angle $\chi > \pi/2$ are anti-consensus configurations. Except for synchronization $\chi = 0$, all these configurations are also balanced. In addition, for $N \geq 4$, irregular consensus and anti-consensus configurations for the undirected ring graph exist where non-consecutive angles of the regular configurations are replaced by $(\pi - \chi)$. This illustrates several properties of consensus and anti-consensus configurations.

- Qualitatively different (anti-)consensus configurations can exist for a particular graph.
- Consensus and anti-consensus configurations can be equivalent when discarding the graph. For example, the positions of 7 agents separated by $2\pi/7$ (consensus) or $4\pi/7$

(anti-consensus) are strictly equivalent, except for *which agent* is located at *which position*; the latter fact is only highlighted by drawing the edges of the graph.
- Degenerate configurations of simultaneous consensus and anti-consensus exist (e.g. $\chi = \pi/2$ for $N = 4, 8, ...$); this singularity seems to be specific to particular graphs.
- There is no common anti-consensus state for all undirected ring graphs. Indeed, considering an agent k, a common anti-consensus state would require that any two other agents, as potential neighbors of k, are either separated by π or located on both sides of k at a distance $\chi \geq \pi/2$; one easily verifies that this cannot be satisfied for all k.

◇

Ex. 5.2.8: balancing on the special orthogonal group: According to Proposition 5.1.5, balancing on $SO(n)$ requires C_e to be a multiple of the identity matrix I_n. Simulations of the algorithms proposed in the next chapter suggest that balanced configurations always exist for $N \geq 2$ if n is even and for $N \geq 4$ if n is odd. Under these conditions, convergence to an anti-consensus configuration different from balancing is not observed for the equally-weighted complete graph. ◇

Ex. 5.2.9: balancing on the Grassmann manifold: Balanced configurations appear on $Grass(p, n)$ when all eigenvalues of C_e are equal. Since $\text{trace}(C_e) = \frac{1}{N}\sum_k \text{trace}(\Pi_k) = p$, this requires $C_e = \frac{p}{n} I_n$. Like for $SO(n)$, simulations tend to indicate that this is feasible when N is large enough; however, computing the minimal value of N for a given n and p seems to be more tricky. ◇

5.3 Consensus as minimizing a cost function

The presence of maximization conditions in the definitions of the previous sections naturally points to the use of optimization methods to compute consensus, anti-consensus and balanced configurations. The present section introduces a cost function whose optimization leads to (anti-)consensus configurations. For a graph $\mathbb{G}$ with adjacency matrix composed of elements a_{jk}, denote the associated in-Laplacian $L^{(i)}$ with elements $l^{(i)}_{jk}$. Consider the variable $x = (x_1, x_2, ..., x_N) \in \mathcal{M}^N$ and define

$$V_L(x) \;=\; \tfrac{1}{2N^2} \sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\, x_j^T x_k \;=\; \Xi_1 - \tfrac{1}{4N^2} \sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\, \|x_j - x_k\|^2 \tag{5.10}$$

with constant $\Xi_1 = \frac{r_{\mathcal{M}}^2}{4N^2} \sum_k \sum_j a_{jk}$. The index L refers to the fact that (5.10) can also be written as a quadratic form on the graph Laplacian:

$$V_L(x) \;=\; \Xi_2 - \tfrac{1}{2N^2} \sum_{k=1}^{N}\sum_{j=1}^{N} l^{(i)}_{jk}\, x_j^T x_k \tag{5.11}$$

with constant $\Xi_2 = \frac{r_{\mathcal{M}}^2}{2N^2} \sum_k d_k^{(i)}$. For the unit-weighted complete graph, $V := V_L + \frac{r_{\mathcal{M}}^2}{2N}$ equals

$$V(x) = \tfrac{1}{2}\|C_e\|^2 \,. \tag{5.12}$$

For the particular case of the circle $\mathcal{M} = S^1$, these cost functions are actually equivalent — up to an additive constant and negative scaling — to V_θ defined by (3.10) in Part I. Thus V_L generalizes, to general connected compact homogeneous manifolds, the cost function used to derive consensus algorithms on the circle.

The choice of the chordal distance $\|x_j - x_k\|$, so far mainly motivated by its computationally simple properties, takes another importance when building V_L. Indeed, defining the cost function V_L with the squared chordal distance ensures that it is smooth everywhere — as the restriction of a smooth function in $\mathbb{R}^n$ to a smooth embedded manifold. This fact is important to get well-behaved algorithms on the basis of the gradient of V_L. It would not hold with the geodesic distance: Figure 5.2 indeed illustrates that the latter is not differentiable everywhere.

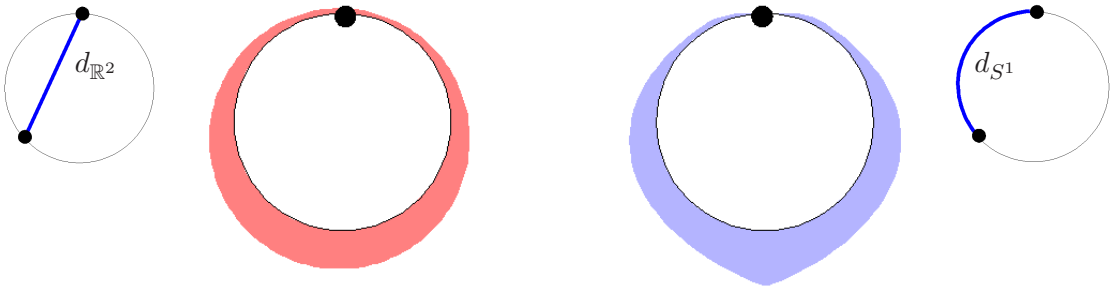


Figure 5.2: Illustrating the smoothness of the squared chordal distance $d^2_{\mathbb{R}^2}$ (left), in opposition to the squared geodesic distance $d^2_{S^1}$ (right) on the circle. The radial thickness of the shaded area above $\theta \in S^1$ represents the distance from θ to the beacon at the top of the circle.

Proposition 5.3.1: *On $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2, synchronization of the N agents is the unique global maximum of V_L whenever the graph $\mathbb{G}$ is weakly connected.*

Proof: According to the second form of (5.10), V_L reaches its global maximum when $x_j = x_k$ for all j, k for which $a_{jk} \neq 0$. If $\mathbb{G}$ is weakly connected, then this equality propagates through the whole graph such that $x_1 = x_2 = ... = x_N$. □

Proposition 5.3.2: *Consider N agents on a manifold $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2. Given an* undirected *graph $\mathbb{G}$, a local maximum of the associated cost function $V_L(x)$ necessarily corresponds to a consensus configuration and a local minimum of $V_L(x)$ necessarily corresponds to an anti-consensus configuration.*

Proof: The proof is given for local maxima; it is strictly analogous for local minima. For $x^* = (x_1^*, x_2^*, ..., x_N^*)$ to be a local maximizer of V_L, x_k^* must be a local maximizer of $V_k(c) := V_L(x_1^*...x_{k-1}^*, c, x_{k+1}^*...x_N^*)$, for each k. V_k takes the linear form $V_k(c) = \Xi_k + \frac{1}{N^2}\, c^T \left(\sum_j a_{jk}\, x_j^*\right)$ since $A = A^T$, with Ξ_k constant $\forall k$. Thanks to Assumption 5.1.2, all local maxima of $V_k(c)$ are global maxima. Therefore, x_k^* is a global maximizer of $V_k(c)$ for all k, which corresponds to Definition 5.2.2 of consensus. □

Proposition 5.3.2 establishes that a *sufficient* condition for (anti-)consensus configurations is to optimize V_L. However, nothing guarantees that this is also *necessary*. In general, optimizing V_L will thus provide proven (anti-)consensus configurations, but not necessarily all of them. This is because, by definition, consensus maximizes V_L on $\mathcal{M}^N$ *for only moving one agent while keeping all others fixed*, and not along directions of combined motion of several agents. The following chapter presents algorithms based on the optimization of V_L to drive the swarm to (anti-)consensus. As a consequence, these algorithms do not necessarily target all possible (anti-)consensus configurations. For instance, for an undirected tree, maximization of V_L always leads to synchronization, although other consensus configurations can exist.

In the frequent case where elements x_k of $\mathcal{M}$ are represented by square matrices, the expression of V_L is as follows:

$$V_L(x) \;=\; \tfrac{1}{2N^2} \sum_{j=1}^{N} \sum_{k=1}^{N} a_{jk} \operatorname{trace}(x_j^T x_k) \qquad \text{with } x_k \in \mathbb{R}^{n\times n} \;\forall k \;. \tag{5.13}$$

Ex. 5.3.3: cost function on the circle: On the circle, V_L is in fact equivalent to V_θ defined by (3.10) in Section 3.2 of Part I. Indeed, identifying $\mathbb{R}^2$ with the complex plane, the second expression of (5.10) for $\mathcal{M} = S^1$ writes $\Xi_1 - \frac{1}{4N^2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\,\|e^{i\theta_j} - e^{i\theta_k}\|^2$, to compare with (3.10) rewritten here for convenience: $V_\theta = \frac{1}{2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\|e^{i\theta_j} - e^{i\theta_k}\|^2$. For more about V_θ, see Section 3.2 in Part I.

In [131], V_θ is used not only to synchronize the headings of agents in the plane, but also to distribute them in balanced configurations. To stabilize *specific* balanced configurations — like the *splay states* where consecutive agents on the circle are separated by a constant angle — with a complete interaction graph, [131] introduces a more complex cost function, involving not only the distance $\|e^{i\theta_k} - e^{i\theta_j}\|$ but also harmonics $\|e^{im\theta_k} - e^{im\theta_j}\|$ with $m > 1$.$\diamond$

Several options have been studied to generalize the complex cost function of [131] to other manifolds, but with no conclusive result so far.

O.Q.: It is currently unclear how to generalize the complex cost functions of [131], involving harmonics of the agents' positions on the circle, to other manifolds in such a way that their extrema correspond to specific configurations.

Ex. 5.3.4: cost function on $SO(n)$: Each term $Q_j^T Q_k = Q_j^{-1} Q_k$ is itself an element of $SO(n)$. It is the unique element of $SO(n)$ translating Q_j to Q_k by matrix (group) multiplication on the right. Hence, on the Lie group $SO(n)$, cost function V_L measures the sum of the traces of the elements translating connected agents to each other. Since the columns and rows of $Q \in SO(n)$ are normalized, the trace is maximal for the identity matrix. One can further imagine how the trace of $Q_j^{-1}Q_k$ characterizes the distance between Q_j and Q_k by examining the case of $SO(2)$, which is actually isometric to the circle: then

$$Q_j^T Q_k = \begin{pmatrix} \cos(\theta_j - \theta_k) & -\sin(\theta_j - \theta_k) \\ \sin(\theta_j - \theta_k) & \cos(\theta_j - \theta_k) \end{pmatrix}$$

where $\theta_j - \theta_k \in S^1$ is the rotation angle between Q_j and Q_k; $\operatorname{trace}(Q_j^T Q_k) = 2\cos(\theta_k - \theta_j)$ indeed monotonically decreases from 2 for $\theta_k = \theta_j$ to -2 for $\theta_k = \theta_j \pm \pi$.

Cost function (5.13) is already used in [22, 97] as a measure of disagreement on $SO(3)$. $\diamond$

Ex. 5.3.5: cost function on Grassmann: On $Grass(p, n)$, (5.13) can be rewritten as

$$V_L(\mathcal{X}) = \frac{1}{2N^2} \sum_{j=1}^{N} \sum_{k=1}^{N} a_{jk} \left(\sum_{i=1}^{p} \cos^2(\phi_{jk}^i) \right)$$

where ϕ_{jk}^i denotes the i^{th} principal angle between $\mathcal{X}_j$ and $\mathcal{X}_k$. This reformulation has previously appeared in [1, 8, 28].

With respect to Example 5.2.9, it can be added that the global minimum of V_L corresponding to balancing is the solution of a simple convex problem *except for* one additional non-convex constraint which contains the whole difficulty. The relaxed convex problem is: minimize $\sum_{j=1}^{n} (\lambda_j)^2$ over $\lambda_j \geq 0$ under the constraint $\sum_{j=1}^{n} (\lambda_j) = p$. The additional constraint is that the λ_j must be the eigenvalues of a matrix C_e which can be written as the sum of N orthonormal projectors of rank p. $\diamond$

Chapter 6

Consensus algorithms on manifolds

The present chapter turns to the design of control algorithms satisfying the fundamental assumptions in this dissertation, i.e. geometric invariance and "autonomous agents" control architecture. The goal is for a swarm of agents on a connected compact homogeneous manifold to converge to the specific configurations defined in Chapter 5. Section 6.1 derives continuous-time gradient algorithms that generalize the algorithm of Section 3.2.2; a discrete-time algorithm is not explicitly provided, but could be associated to V_L along the same lines as in Chapter 3. Section 6.2 generalizes to the present setting the controllers of Chapter 4 for global synchronization. Section 6.2.1 "cheats" the non-convexity of the manifold structure with auxiliary variables to obtain almost-global synchronization for uniformly connected graphs, as in Section 4.3; a similar balancing algorithm is also proposed. Section 6.2.2 briefly discusses the generalization of the modified interaction profile of Section 4.1, and Section 6.2.3 reexamines the Gossip Algorithm of Section 4.2.

6.1 Gradient consensus algorithms

The formulation of (anti-)consensus, synchronization and balancing as extrema of cost functions V_L and V in Section 5.3 paves the way for the design of ascent and descent algorithms to reach these configurations. The following considers continuous-time gradient algorithms, but any descent or ascent algorithm — in particular, discrete-time — will achieve the same task; see [2] for extensive information on this subject.

In the present context, as in the rest of the dissertation, the gradient is always defined with the canonical metric induced by the embedding of $\mathcal{M}$ in $\mathbb{R}^m$, see Section 2.4. Formally, a gradient algorithm for V_L can be written

$$\frac{d}{dt}x_k = 2N^2\,\alpha\ \mathrm{grad}_{k,\mathcal{M}}(V_L)\ , \qquad k = 1, 2, ..., N\ , \tag{6.1}$$

with $\alpha > 0$ (resp. $\alpha < 0$) for consensus (resp. anti-consensus), and where $\mathrm{grad}_{k,\mathcal{M}}(f)$ denotes the gradient of f with respect to x_k along $\mathcal{M}$. This gradient can be obtained, as outlined in Section 2.4, from the gradient in $\mathbb{R}^m$

$$\mathrm{grad}_{k,\mathbb{R}^m}(V_L) = \tfrac{1}{2N^2}\ \textstyle\sum_j (a_{jk} + a_{kj})\ x_j$$

by orthogonal projection $\text{Proj}_{T\mathcal{M},k}$ onto the tangent space to $\mathcal{M}$ at x_k, yielding

$$\begin{aligned} \tfrac{d}{dt}x_k &= \alpha \, \text{Proj}_{T\mathcal{M},k}\left(\textstyle\sum_j (a_{jk}+a_{kj})x_j\right) \\ &= \alpha \, \text{Proj}_{T\mathcal{M},k}\left(\textstyle\sum_j (a_{jk}+a_{kj})(x_j-x_k)\right) \ , \qquad k=1,2,...,N \ . \end{aligned} \tag{6.2}$$

The last equality comes from the property $\text{Proj}_{T\mathcal{M},k}(x_k) = 0$. It shows that to implement this consensus algorithm, each agent k must know the relative position with respect to itself of all agents j such that $j \rightsquigarrow k$ or $k \rightsquigarrow j$. This is in agreement with the required geometric invariance properties. However, regarding inter-agent communication, since the information flow is restricted to $j \rightsquigarrow k$, (6.2) can only be implemented for undirected graphs, for which it becomes

$$\tfrac{d}{dt}x_k = 2\alpha \, \text{Proj}_{T\mathcal{M},k}\left(\sum_{j=1}^{N} a_{jk}(x_j - x_k)\right) \ , \qquad k=1,2,...,N \ . \tag{6.3}$$

In the special case of a complete unit-weighted graph,

$$\tfrac{d}{dt}x_k = 2\alpha N \, \text{Proj}_{T\mathcal{M},k}\left(C_e(t) - x_k\right) \ , \qquad k=1,2,...,N \tag{6.4}$$

which shows that each agent moves towards the *IAM* of the swarm's current positions. It is straightforward to see that for the special case of the circle $\mathcal{M} = S^1$, expression (6.3) becomes algorithm (3.13) introduced in Part I.

Proposition 6.1.1: *Consider a swarm of N agents moving according to (6.3) on a manifold $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2, with undirected graph $\mathbb{G}$ associated to the a_{jk}. This swarm always converges to a set of equilibrium points that consists of the critical points of V_L. If $\alpha < 0$, all asymptotically stable equilibria are anti-consensus configurations for $\mathbb{G}$. If $\alpha > 0$, all asymptotically stable equilibria are consensus configurations for $\mathbb{G}$; for the equally-weighted complete graph and for tree graphs, the only asymptotically stable configuration is synchronization).*

Proof: $\mathcal{M}$ being compact and the a_{jk} bounded, V_L is upper- and lower-bounded. For $\alpha > 0$ ($\alpha < 0$), V_L is always increasing (decreasing) along solutions of (6.3), since

$$\tfrac{d}{dt}V_L \ = \ \textstyle\sum_k \, (\tfrac{d}{dt}x_k)^T \, \text{grad}_{k,\mathcal{M}}(V_L) \ = \ 2N^2\alpha \ \sum_k \|\text{grad}_{k,\mathcal{M}}(V_L)\|^2 \ .$$

By LaSalle's invariance principle, the swarm converges towards a set where $\frac{d}{dt}V_L = 0$, implying $\text{grad}_{k,\mathcal{M}}(V_L) = 0 \ \Leftrightarrow \ \frac{d}{dt}x_k = 0 \ \forall k \in \mathcal{V}$ which characterizes a set of equilibria. For $\alpha > 0$ ($\alpha < 0$), since V_L always increases (decreases) along solutions, only its local maxima (minima) can be asymptotically stable. This concludes the general proof since Proposition 5.3.2 states that all local maxima (minima) of V_L correspond to consensus (anti-consensus) configurations. For the complete graph, consensus means synchronization from Proposition 5.2.4; for the tree, it is not difficult to show that synchronization is the only local maximum of V_L, working along the lines of the proof of Proposition 3.3.6. □

Formally, algorithm (6.3) can be written for directed and time-varying graphs, although the gradient property is lost for directed graphs and has little meaning in the time-varying

case (since V_L then explicitly depends on time). Nevertheless, the general case of (6.3) still exhibits synchronization properties. In particular, the following generalizes Proposition 3.3.1.

Proposition 6.1.2: *Consider a swarm of N agents moving according to (6.3) with $\alpha > 0$ on a manifold $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2, where the graph $\mathbb{G}$ associated to the a_{jk} is uniformly connected. If the initial positions of the agents $x_k(0)$, $k = 1, 2, ..., N$, are located within a convex set of $\mathcal{M}$, then the swarm converges to synchronization.*

Proof: The intuition behind the proof is the same as for Proposition 3.3.1, but the formal argument can be made along more general lines of [95]: for any dynamical system satisfying the proper convexity property, the convex hull of the agents' positions shrinks until synchronization is reached. The convexity condition is satisfied in the present context if the agents are initially located within a convex set of $\mathcal{M}$. □

On the other hand, the examples of Section 3.4 illustrate that global convergence properties of (6.3) are unclear even for the simple case of the circle when $\mathbb{G}$ is not fixed and undirected. Simulations on $SO(n)$ and $Grass(p, n)$ seem to indicate that for randomly generated digraph sequences[1], the swarm eventually converges to synchronization when $\alpha > 0$; this would correspond to *generic* convergence for unconstrained graphs. The behavior for $\alpha < 0$ is unclear with varying graphs; this is not surprising since there exists no common equilibrium for all graphs, except synchronization which is unstable when $\alpha < 0$.

The following sections consider alternative algorithms — generalized from Chapter 4 — whose purpose is to enhance the proven convergence properties of the present section.

Ex. 6.1.3: gradient consensus on S^1: Particularizing expression (6.3) to the circle directly leads to algorithm (3.13) which is a fundamental element of Part I of this dissertation. Convergence results obtained for (3.13) in Section 3.3 all have their counterparts on general manifolds in Section 6.1. In particular, the definitions of consensus and anti-consensus introduced in Section 5.2 provide formal characterizations for the "spurious local minima of V_θ" to which algorithm (3.13) can converge for some $\mathbb{G}$. In return, all the examples and counterexamples of Chapter 3 can be consulted to illustrate properties of (6.3). ⋄

Ex. 6.1.4: gradient consensus on $SO(n)$: The tangent space to $SO(n)$ at the identity I_n is the space of antisymmetric $n \times n$ matrices. By group multiplication, the projection of $B \in \mathbb{R}^{n\times n}$ onto the tangent space to $SO(n)$ at Q_k is $Q_k\,(\frac{Q_k^T B}{2} - \frac{B^T Q_k}{2})$ (see Section 2.4). This leads to the following explicit form of algorithm (6.3) on $SO(n)$, where the right-hand side only depends on relative positions of the agents:

$$Q_k^T \tfrac{d}{dt} Q_k = \alpha \sum_{j=1}^{N} a_{jk} \left(Q_k^T Q_j - Q_j^T Q_k\right) , \qquad k = 1, 2, ..., N . \tag{6.5}$$

Proposition 6.1.5: *The manifold $SO(n)$ satisfies Assumption 5.1.2.*

Proof: (+ Proof of Proposition 5.1.5) Consider a linear function $f(Q) = \text{trace}(Q^T B)$ with $Q \in SO(n)$ and $B \in \mathbb{R}^{n\times n}$. Then $\text{grad}_{\mathbb{R}^{n\times n}}(f) = B$ so $\text{grad}_{SO(n)}(f) = \frac{Q}{2}(Q^T B - B^T Q)$.

[1] More precisely, the following distribution is examined. Initially, each element a_{jk} independently takes a value in $\{0, 1\}$ according to a probability $\text{Prob}(1) = p$. The corresponding graph remains for a time t_{graph} uniformly distributed in $[t_{min}, t_{max}]$, after which a new graph is built as initially.

Since Q is invertible, critical points of f satisfy $(Q^T B - B^T Q) = 0$, meaning that they take the form described by Lemma A.1 in the Appendix. Using notations of Lemma A.1, write $R = H\Lambda H^T$ where Λ contains the (non-negative) eigenvalues of R. This leads to

$$Q = UHJH^T \quad \Rightarrow \quad Q^T B = HJ\Lambda H^T \quad \Rightarrow \quad f(Q) = -\sum_{j=1}^{l} \Lambda_{jj} + \sum_{j=l+1}^{n} \Lambda_{jj} \ .$$

If $l \geq 2$, select any $m \in [2, l]$ and define $Q_\varepsilon = UHJAH^T$ where A is the identity matrix except that $A(1,1) = A(m,m) = \cos(\varepsilon)$ and $A(1,m) = -A(m,1) = \sin(\varepsilon)$ with ε arbitrarily small. Then $f(Q_\varepsilon) > f(Q)$ unless $\Lambda_{11} = \Lambda_{mm} = 0$. Similarly, if $l = 1$ and $\exists\ m \geq 2$ such that $\Lambda_{mm} < \Lambda_{11}$, then $f(Q_\varepsilon) > f(Q)$ with Q_ε and A defined as previously. Note that J_{ss} does not affect the value of $f(Q)$ when $\Lambda_{ss} = 0$; therefore these cases can be discarded and

- if $\det(B) \geq 0$, then local maxima require $l = 0$, such that $Q = U$ and $f(Q)$ is the sum of the eigenvalues of R;
- if $\det(B) \leq 0$, then local maxima require U to take the form of Lemma A.1 with $l = 1$ and $\Lambda_{11} \leq \Lambda_{mm} \forall m$; thus the first column of H corresponds to a smallest eigenvalue of R and $f(Q)$ is the sum of $n-1$ largest eigenvalues minus the smallest one.

This shows that all maxima of $f(Q)$ are global maxima (since they all take the same value). Moreover, taking $B = C_e$, it characterizes the *IAM* as described in Proposition 5.1.5. □

Example 5.3.3 mentions that [131] stabilizes more specific configurations like splay states on the circle by using a cost function that combines distances of harmonics of the relative positions. On Lie groups like $SO(n)$, one could imagine to define harmonics as consecutive group products $Q_j^T Q_k$, $(Q_j^T Q_k)^2$, $(Q_j^T Q_k)^3$,... . However, simulations of cost functions similar to those of [131] derived in this way for $SO(3)$ have led to no conclusive result. ⋄

Ex. 6.1.6: gradient consensus on $Grass(p,n)$: The projection of a matrix $M \in \mathbb{S}_n^+$ onto the tangent space to $Grass(p,n)$ at $\mathcal{X}_k$ represented by Π_k is given in [79] as $\Pi_k M \Pi_{\perp k} + \Pi_{\perp k} M \Pi_k$. This leads to

$$\tfrac{d}{dt}\Pi_k = 2\alpha \sum_{j=1}^{N} a_{jk} \left(\Pi_k \Pi_j \Pi_{\perp k} + \Pi_{\perp k} \Pi_j \Pi_k\right) , \qquad k = 1, 2, ..., N \ . \tag{6.6}$$

In practice, the basis representation X_k of $\mathcal{X}_k \in Grass(p,n)$ is handier than Π_k, because it involves smaller matrices. Computing the gradient of $V_L(\{\Pi_k : k = 1, 2, ..., N\}) = V_L(\{X_k\, X_k^T : k = 1, 2, ..., N\})$ directly on the quotient manifold as explained in [1] leads to the algorithm

$$\tfrac{d}{dt}X_k = 4\alpha \sum_{j=1}^{N} a_{jk} \left(X_j\, M_{j\cdot k} - X_k\, M_{j\cdot k}^T M_{j\cdot k}\right) , \qquad k = 1, 2, ..., N \ , \tag{6.7}$$

where the $p \times p$ matrices $M_{j\cdot k}$ are defined as $M_{j\cdot k} = X_j^T X_k$. Algorithm (6.7) allows to get the same trajectory on $Grass(p,n)$ as with algorithm (6.6) but using matrices $X_k \in \mathbb{R}^{n\times p}$; this allows to save significant memory and computation time. However, the projector representation is a necessary choice for theoretical purposes, since the formalism is developed for embedded manifolds and the basis representation is not an embedding in $\mathbb{R}^{n\times p}$.

Proposition 6.1.7: *The Grassmann manifold satisfies Assumption 5.1.2.*

Proof: (+ Proof of Proposition 5.1.7) Consider a linear function $f(\Pi) = \text{trace}(\Pi^T B)$ where $B \in \mathbb{S}_n^+$ and Π represents $\mathcal{X} \in Grass(p,n)$. Then $\text{grad}_{\mathbb{R}^{n\times n}}(f) = B$ and $\text{grad}_{Grass(p,n)}(f) =$

$\Pi B \Pi_\perp + \Pi_\perp B \Pi$. The ranges of the first and second terms in $\text{grad}_{Grass(p,n)}(f)$ are at most $\mathcal{X}$ and its orthogonal complement respectively, so they must both equal zero at a critical point $\mathcal{X}^*$, such that $\mathcal{X}^*$ is an invariant subspace of B. In an appropriate orthonormal basis $(\mathbf{e}_1, \mathbf{e}_2, ..., \mathbf{e}_n)$ of $\mathbb{R}^n$, write $\Pi^* = \text{diag}(1, ..., 1, 0, ..., 0)$ and $B = \text{diag}(\mu_1, ..., \mu_p, \mu_{p+1}, ..., \mu_n)$. If $\exists\ d \leq p$ and $l > p$ such that $\mu_d < \mu_l$, then any variation of Π^* rotating a vector initially aligned with $\mathbf{e}_d$ towards $\mathbf{e}_l$ strictly increases $f(\Pi)$. Therefore, at local maxima of $f(\Pi)$, the p-dimensional space corresponding to Π must be an eigenspace of B corresponding to p largest eigenvalues. This implies that any local maximum of $f(\Pi)$ equals the sum of p largest eigenvalues of B, so Assumption 5.1.2 is satisfied. Replacing B by C_e characterizes the *IAM* as described in Proposition 5.1.7. □

◇

6.2 Global consensus on manifolds

Section 6.1 proposes algorithms that lead to a consensus situation which is linked to the particular interconnection graph $\mathbb{G}$. In many applications however, the interconnection graph is just a restriction on communication possibilities, under which one actually wants to achieve a consensus for the complete graph (i.e. synchronization or balancing). Moreover, allowing directed and time-varying communication graphs is desirable for robustness.

Chapter 4 presents algorithms for global synchronization on the circle: modifying the attraction profile among interconnected agents, randomly selecting existing communication links, and using auxiliary variables with a dynamic controller. The present section discusses their extension to connected compact homogeneous manifolds and to the task of balancing.

6.2.1 Dynamic controller with auxiliary variables

The algorithms of the present section are derived as an extension of the dynamic controller presented on the circle in Section 4.3. They achieve the same performance as those of Section 6.1 for the equally-weighted complete graph — that is, driving the swarm to synchronization or to a subset of the anti-consensus configurations of the equally-weighted complete graph — under the weaker condition of uniform connectedness on the actual interconnection graph $\mathbb{G}$. The reduction of information channels is compensated for by adding a consensus variable $y_k \in \mathbb{R}^m$ to the state space of each agent. This is the strict generalization of the auxiliary variable in $\mathbb{R}^2 \cong \mathbb{C}$ used for synchronization on the circle in Section 4.3. The agents must not only memorize and update their auxiliary variable, but also communicate it to their out-neighbors; thus the proposed strategy is feasible for engineering applications where variables can be communicated among agents, but it is certainly a questionable model to describe natural phenomena.

The presentation focuses on continuous-time algorithms to avoid lengthy discussion; a discrete-time adaptation of the algorithm for balancing is illustrated on the circle.

Synchronization algorithm. For synchronization purposes, as in Section 4.3, the agents run a classical consensus algorithm on their estimator variables y_k in $\mathbb{R}^m$, $k = 1, 2, ..., N$, to "cheat" the non-convexity of the actual manifold $\mathcal{M}$. The auxiliary variables are initialized arbitrarily but independently and such that they can take any value in an open subset of $\mathbb{R}^m$.

In addition, $\forall k \in \mathcal{V}$, agent k's position x_k on $\mathcal{M}$ independently tracks (the projection on $\mathcal{M}$ of) y_k. This leads to

$$\frac{d}{dt}y_k = \beta \sum_{j=1}^{N} a_{jk}\,(y_j - y_k) \quad , \quad \beta > 0 \tag{6.8}$$

$$\frac{d}{dt}x_k = \gamma_S\, \mathrm{grad}_{k,\mathcal{M}}(x_k^T\, y_k) = \gamma_S\, \mathrm{Proj}_{T\mathcal{M},k}(y_k) \quad , \quad \gamma_S > 0\,, \quad k = 1, 2, ..., N. \tag{6.9}$$

The projection of $y \in \mathbb{R}^m$ onto $\mathcal{M}$ is defined by $\mathrm{Proj}_{\mathcal{M}}(y) := \mathrm{argmax}_{c\in\mathcal{M}}(c^T y)$, which is equivalent to the classical definition $\mathrm{Proj}_{\mathcal{M}}(y) = \mathrm{argmin}_{c\in\mathcal{M}}\|c - y\|$ as $\|c\| = r_{\mathcal{M}}\ \forall c \in \mathcal{M}$.

The geometric invariance of algorithms (6.8) and (6.9) must be examined. According to Assumption 5.0.2, the symmetry group of $\mathcal{M}$ is a subgroup of the orthogonal group $O(m)$. Regarding (6.9), consider its first expression: the function $(x_k^T\, y_k)$ is invariant under the change of variables $(x_1, x_2, ..., x_N,\ y_1, y_2..., y_N) \to (Qx_1, Qx_2, ..., Qx_N,\ Qy_1, Qy_2, ..., Qy_N)$ for any $Q \in O(m)$, and invariance of its gradient directly follows from the fact that Q must be in the symmetry group of $\mathcal{M}$. In (6.8), only relative values $y_j - y_k$ of the auxiliary variables appear and it is clear that the algorithm is invariant under the change of variables $(y_1, y_2..., y_N) \to (Qy_1, Qy_2, ..., Qy_N)$ for any $Q \in O(m)$. This confirms that the algorithm preserves the geometric invariance of $\mathcal{M}$. A remaining issue is in which form autonomous agents with no common reference can exchange auxiliary variables y_k; that is, is it possible to rewrite (6.8),(6.9) with auxiliary variables which are *relative* to the agents' positions, like it is done on the circle by transforming (3.1),(3.3),(4.5) into (4.7) ? This question is not so easy to answer on more general manifolds and it is discussed separately at the end of the present section, since it changes nothing to the behavior of the algorithms.

The following Proposition assesses the convergence property of the complete algorithm (6.8),(6.9). Notation IAM_g is used to generalize the definition (5.2) of the *IAM* when the points defining C_e are not on $\mathcal{M}$.

Proposition 6.2.1: *Consider a piecewise continuous and uniformly connected graph $\mathbb{G}(t)$ and a manifold $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2. Consider a set of N agents evolving on $\mathcal{M}$ under (6.8),(6.9), with the y_k initialized arbitrarily but independently and such that they can take any value in an open subset of $\mathbb{R}^m$. Then for all initial conditions $x_k(0)$, $y_k(0)$, $k = 1, 2, ..., N$, the solutions converge to a set of equilibria. Moreover, all* stable[2] *equilibria correspond to situations with the y_k synchronized at some $\bar{y}$ for which $\mathrm{Proj}_{\mathcal{M}}(\bar{y})$ reduces to a unique point, and with the agent positions x_k synchronized at $\bar{x} = \mathrm{Proj}_{\mathcal{M}}(\bar{y})$; if $\mathbb{G}(t)$ is balanced, then $\bar{x} = IAM_g\{y_k(0) : k = 1, 2..., N\}$.*

Proof: Propositions 3.1.1 and 3.1.2, summarizing results of [94, 95, 104], imply convergence of (6.8) towards $y_k = \bar{y}\ \forall k \in \mathcal{V}$ as well as the associated property $\bar{y} = \frac{1}{N}\sum_k y_k(0)$ for balanced graphs. Therefore, algorithm (6.8),(6.9) is an asymptotically autonomous system where in the limiting autonomous system, $y_k = \bar{y}$ is constant $\forall k \in \mathcal{V}$. As a consequence, the asymptotic form of (6.8),(6.9) reduces to a set of N independent equations

$$\frac{d}{dt}x_k = \gamma_S\, \mathrm{Proj}_{T\mathcal{M},k}(\bar{y})\,, \qquad k = 1, 2, ..., N\,. \tag{6.10}$$

This is a gradient ascent algorithm for $f(x_k) = x_k^T \bar{y}$. According to the result of [91] recalled in Proposition 2.3.6, the limit set of the original system (6.8),(6.9) corresponds to the chain

[2]This stability must be understood as Lyapunov stability with respect to translations of the x_k and y_k along the invariance directions of the system (trivial) and asymptotic stability in the other directions.

recurrent set of the asymptotic system (6.10). Moreover, according to Proposition 2.3.8 and since $f(x_k)$ is smooth (as the restriction of a smooth function to the smooth embedded manifold $\mathcal{M}$), the chain recurrent set of (6.10) is equal to its equilibrium set. This proves that all solutions converge to a set of equilibria where x_k is at a critical point of $f(x_k)$ $\forall k \in \mathcal{V}$.

If $\bar{y}$ is such that $f(x_k)$ has a unique maximizer x_*, then the latter is the only stable equilibrium for gradient ascent algorithm (6.10) with respect to variations of x_k. Therefore, the only stable situation is $x_k = x_*$ $\forall k \in \mathcal{V}$, implying synchronization. The proof concludes by showing that situations with $\bar{y}$ such that $f(x_k)$ has multiple maximizers on $\mathcal{M}$ are unstable with respect to variations of the y_k.

Continuous variations of the y_k affect $f(x_k)$ as a continuous variation of $\bar{y}$, so stability is examined with respect to variations of $\bar{y}$. Consider an arbitrary point $\bar{y}_o \in \mathbb{R}^m$. If $x_k^T \bar{y}_o$ has multiple maximizers, then select one of them, call it x_*. For arbitrarily small $\sigma \in \mathbb{R}_{>0}$, define $\bar{y}_u = \bar{y}_o + \sigma x_*$. Then $x_k^T \bar{y}_u \leq x_*^T \bar{y}_u$ with equality holding if and only if $x_k = x_*$. Thus the situation with multiple maximizers is unstable at least in the direction of x_*. □

With an a priori arbitrary choice of the $y_k(0)$, it could well be that, even if the x_k are initially close, they are in fact driven away to a completely different place $\bar{x} = \mathrm{Proj}_{\mathcal{M}}(\bar{y})$. A choice like e.g. $y_k(0) = \rho_k x_k(0)$ with $\rho_k \in \mathbb{R}_{>0}$ $\forall k$ can avoid unnecessary transients when the $x_k(0)$ are close.

Anti-consensus algorithm. For anti-consensus, in analogy with the synchronization strategy, each x_k evolves according to a gradient algorithm to *maximize* its distance to its associated auxiliary variable $y_k(t)$. If the $y_k(t)$ asymptotically converge to $C_e(t)$, this becomes equivalent to the gradient anti-consensus algorithm (6.4). Imposing $y_k(0) = x_k(0)$ $\forall k$, the following algorithm achieves this purpose when the graph $\mathbb{G}(t)$ is balanced $\forall t$:

$$\frac{d}{dt} y_k \;=\; \beta \sum_{j=1}^{N} a_{jk} \, (y_j - y_k) \;+ \frac{d}{dt} x_k \qquad , \quad \beta > 0 \tag{6.11}$$

$$\frac{d}{dt} x_k \;=\; \gamma_B \; \mathrm{grad}_{k,\mathcal{M}}(x_k^T y_k) \;=\; \gamma_B \, \mathrm{Proj}_{T\mathcal{M},k}(y_k) \qquad , \quad \gamma_B < 0 \, , \quad k = 1, 2, ..., N. \tag{6.12}$$

The invariance properties of (6.11),(6.12) are as for (6.8),(6.9). Variables x_k and y_k are fully coupled in (6.11),(6.12). In a discrete-time version of the system, this essential feature of the algorithm must be retained in the form of *implicit* update equations in order to ensure convergence. Example 6.2.3 illustrates this fact on the circle.

Proposition 6.2.2: *Consider a piecewise continuous, uniformly connected and balanced graph $\mathbb{G}(t)$ and a manifold $\mathcal{M}$ satisfying Assumptions 5.0.2 and 5.1.2. Then, algorithm (6.11),(6.12) with initial conditions $y_k(0) = x_k(0)$ $\forall k \in \mathcal{V}$ converges to an equilibrium configuration of the anti-consensus algorithm for the equally-weighted complete graph, that is of (6.4) with $\alpha < 0$.*

Proof: It is first shown that $\frac{1}{N}\sum_k y_k(t) = \frac{1}{N}\sum_k x_k(t) = C_e(t)$ $\forall t \geq 0$. Since $y_k(0) = x_k(0)$ $\forall k \in \mathcal{V}$, it is true for $t = 0$. Thus it remains to show that $\sum_k \frac{d}{dt} y_k(t) = \sum_k \frac{d}{dt} x_k(t)$. This is the case because a balanced graph $\mathbb{G}$ ensures that the first two terms on the right side of the following expression cancel each other:

$$\textstyle\sum_k \frac{d}{dt} y_k(t) \;=\; \beta \sum_j \left(\sum_k a_{jk}\right) y_j \;-\; \beta \sum_k \left(\sum_j a_{jk}\right) y_k \;+\; \sum_k \frac{d}{dt} x_k(t) \, .$$

Next, it is proven that $\forall k \in \mathcal{V}$, $\frac{d}{dt}x_k(t)$ is a uniformly continuous function of t in $L_2(0,+\infty)$; then Barbalat's Lemma, recalled in Proposition 2.3.1, implies that $\frac{d}{dt}x_k$ goes to 0 as t goes to $+\infty$, meaning that the x_k asymptotically approach an equilibrium set. First observe that $W(t) := \frac{1}{2}\sum_k y_k(t)^T y_k(t)$ is never increasing along the solutions of (6.11),(6.12). Indeed, denoting by $(y)_j,\ j = 1,2,...,m$, the vectors of length N containing the j-th component of every y_k, $k = 1,2,...,N$, and by $L(t)$ the Laplacian of $\mathbb{G}(t)$ (since $\mathbb{G}(t)$ is balanced, $L(t) = L^{(i)}(t) = L^{(o)}(t)$), one obtains

$$\frac{d}{dt}W(t) \;=\; \sum_k y_k^T \frac{d}{dt}y_k \;=\; \sum_k y_k^T \frac{d}{dt}x_k \;-\; \beta\ \sum_j (y)_j^T L\,(y)_j\ .$$

The Laplacian of balanced graphs is positive semidefinite (see [149]). Replacing $\frac{d}{dt}x_k$ from (6.12) and noting that $y_k^T\ \mathrm{Proj}_{T\mathcal{M},k}(y_k) \;=\; \big(\mathrm{Proj}_{T\mathcal{M},k}(y_k)\big)^T\ \mathrm{Proj}_{T\mathcal{M},k}(y_k)$, one obtains

$$\frac{d}{dt}W(t) \;=\; \gamma_B \sum_{k=1}^{N} \|\mathrm{Proj}_{T\mathcal{M},k}(y_k)\|^2 \;-\; \beta \sum_{j=1}^{N} (y)_j^T L\,(y)_j \leq 0\ . \tag{6.13}$$

Thus $W(t) \leq W(0) = \frac{N}{2}\,r_{\mathcal{M}}^2$. This implies that each $\frac{d}{dt}x_k(t)$ is a function of t in $L_2(0,+\infty)$ since

$$\tfrac{1}{|\gamma_B|}\textstyle\sum_k \int_0^{+\infty} \|\frac{d}{dt}x_k(t)\|^2\,dt \;\leq\; -\int_0^{+\infty} \frac{d}{dt}W(t)\,dt \;\leq\; \frac{N}{2}\,r_{\mathcal{M}}^2\ .$$

$W(t) \leq W(0)$ also implies that each y_k is uniformly bounded. From (6.12), $\frac{d}{dt}x_k$ is uniformly bounded as well. Combining these two observations, with the a_{jk} bounded, (6.11) shows that each y_k has a bounded derivative and hence is Lipschitz in t. Now write

$$\|\tfrac{d}{dt}x_k(x_k(t_1),y_k(t_1)) - \tfrac{d}{dt}x_k(x_k(t_2),y_k(t_2))\| \;\leq$$

$$\|\tfrac{d}{dt}x_k(x_k(t_1),y_k(t_1)) - \tfrac{d}{dt}x_k(x_k(t_1),y_k(t_2))\| + \|\tfrac{d}{dt}x_k(x_k(t_1),y_k(t_2)) - \tfrac{d}{dt}x_k(x_k(t_2),y_k(t_2))\|\ .$$

The first term on the second line is bounded by $r_1\,|t_1 - t_2|$ for some r_1 since $\frac{d}{dt}x_k$ is linear in y_k and y_k is Lipschitz in t. The second term on the second line is bounded by $r_2\,|t_1 - t_2|$ for some r_2 since $\frac{d}{dt}x_k$ is Lipschitz in x_k (as the gradient of a smooth function along the smooth manifold $\mathcal{M}$) and $\frac{d}{dt}x_k$ is uniformly bounded. Hence, $\frac{d}{dt}x_k$ is Lipschitz in t and therefore uniformly continuous in t, such that Barbalat's Lemma can be applied.

According to the result of [91] recalled in Proposition 2.3.6, the limit set of the original system (6.8),(6.9) corresponds to the chain recurrent set of the asymptotic system

$$\frac{d}{dt}y_k \quad = \quad \beta\ \textstyle\sum_j a_{jk}\,(y_j - y_k)$$

resulting when the $\frac{d}{dt}x_k$ go to 0, associated with the condition $\mathrm{Proj}_{T\mathcal{M},k}(y_k) = 0$ for the x_k. The chain recurrent set of the linear consensus algorithm in the asymptotic system reduces to its equilibrium set $y_k = \bar{y}\ \forall k \in \mathcal{V}$. But then, from the beginning of the proof, $y_k = C_e$ $\forall k \in \mathcal{V}$ such that the static condition for the x_k becomes $\mathrm{Proj}_{T\mathcal{M},k}(C_e) = 0\ \forall k \in \mathcal{V}$. This is the condition for an equilibrium of anti-consensus algorithm (6.4) for the complete graph and with $\gamma_B = 2\alpha N$, which concludes the proof. □

In simulations, a swarm applying (6.11),(6.12) with $y_k(0) = x_k(0)\ \forall k$ seems to generically converge to an anti-consensus configuration of the equally-weighted complete graph, that is a *stable* equilibrium configuration of (6.4) with $\alpha < 0$.

Ex. 6.2.3: consensus with auxiliary variables on S^1: As in Part I, auxiliary variable y_k is assumed to belong to $\mathbb{R}^2$ or to $\mathbb{C}$ according to the best convenience for notation. Particularizing (6.8),(6.9) to the circle yields the dynamic controller for synchronization introduced in Section 4.3: $x_k =: e^{i\theta_k}$ tracks the central projection of y_k onto the unit circle. The change of variables $r_k = y_k e^{-i\theta_k}$ allows to express the whole algorithm with agent-centered auxiliary variables, which gives the controller (4.7) rewritten here for convenience

$$\begin{aligned} \tfrac{d}{dt}\theta_k &= \gamma_S\, \rho_k\, \sin(\chi_k) \quad , \qquad \gamma_S > 0 \\ \tfrac{d}{dt} r_k &= \beta \sum_{j=1}^{N} a_{jk} \left(r_j e^{i(\theta_j - \theta_k)} - r_k \right) - i r_k \tfrac{d}{dt}\theta_k \quad , \qquad \beta > 0 \end{aligned} \tag{6.14}$$

$$\text{with complex polar representation } r_k = \rho_k e^{i\chi_k} \qquad , \quad k = 1, 2, ..., N \; .$$

The balancing algorithm (6.11),(6.12) writes

$$\begin{aligned} \tfrac{d}{dt}\theta_k &= \gamma_B\, y_k^T q_k \\ \tfrac{d}{dt} y_k &= \beta \sum_{j=1}^{N} a_{jk}\, (y_j - y_k) + q_k\, \tfrac{d}{dt}\theta_k \end{aligned}$$

where q_k is the vector in $\mathbb{R}^2$ corresponding to $ie^{i\theta_k} \in \mathbb{C}$. The same change of variables as before $r_k = y_k e^{-i\theta_k}$ allows to express the balancing algorithm with agent-centered auxiliary variables, yielding

$$\begin{aligned} \tfrac{d}{dt}\theta_k &= \gamma_B\, \rho_k\, \sin(\chi_k) \quad , \qquad \gamma_B < 0 \\ \tfrac{d}{dt} r_k &= \beta \sum_{j=1}^{N} a_{jk} \left(r_j e^{i(\theta_j - \theta_k)} - r_k \right) - i(r_k - 1) \tfrac{d}{dt}\theta_k \quad , \qquad \beta > 0 \end{aligned} \tag{6.15}$$

$$\text{with complex polar representation } r_k = \rho_k e^{i\chi_k} \qquad , \quad k = 1, 2, ..., N$$

and where $r_k(0) = 1e^{i0}\ \forall k \in \mathcal{V}$ to have $y_k = x_k$ as required by Proposition 6.2.2.

The discrete-time algorithm with the same convergence properties as (6.15) is

$$\begin{aligned} \theta_k(t+1) &= \theta_k(t) + \arg\left((1-\delta) - \delta\, r_k(t+1) e^{i(\theta_k(t+1) - \theta_k(t))} \right) \\ r_k(t+1) &= 1 + \left(\beta \sum_{j=1}^{N} a_{jk} \left(r_j(t) e^{i(\theta_j(t) - \theta_k(t))} - r_k(t) \right) + r_k(t) - 1 \right) e^{-i(\theta_k(t+1) - \theta_k(t))} \end{aligned} \tag{6.16}$$

$$, \quad k = 1, 2, ..., N$$

with $\delta \in (0, 1)$, gain $\beta > 0$ and, as previously, $r_k(0) = 1e^{i0}\ \forall k \in \mathcal{V}$. As previously announced, (6.16) is an *implicit* update equation where θ_k and r_k are fully coupled: the new value of θ_k depends on the new value of r_k (first line of (6.16)) and similarly the new value of r_k depends on the new value of θ_k (second line of (6.16)), in a nonlinear way that makes it nontrivial to decouple the update equations. Unfortunately, this seems to be necessary to obtain the same convergence as for (6.15).

Proposition 6.2.4: *Consider a uniformly connected and balanced graph $\mathbb{G}(t)$. Consider a swarm of N agents moving on the circle according to algorithm (6.16), with $\delta_k \in (0,1)$, gain $\beta \in (0, \frac{1}{d_{max}})$ where $d_{max} := \max_{k\in\mathcal{V}} (d_k^{(i)})$ is the maximal in-degree in graph $\mathbb{G}$, and $r_k(0) = 1e^{i0}\ \forall k \in \mathcal{V}$. Then the swarm converges to an equilibrium configuration of the anti-consensus algorithm for the equally-weighted complete graph.*

Proof: The dynamics of $y_k = r_k e^{i\theta_k}$ satisfy

$$y_k(t+1) = y_k(t) + \beta \sum_{j=1}^{N} a_{jk}\ (y_j(t) - y_k(t)) + e^{i\theta_k(t+1)} - e^{i\theta_k(t)}\ , \qquad k = 1, 2, ..., N\,.$$

As for the proof of Proposition 6.2.2 in continuous-time, define $W = \frac{1}{2}\ \sum_k\ y_k^T\, y_k$, where $y_k \in \mathbb{R}^2$. Defining $y \in \mathbb{C}^N$ as the complex vector containing all the $y_k \in \mathbb{C}$, rewrite

$$y_k(t) + \beta \sum_{j=1}^{N} a_{jk}\ (y_j(t) - y_k(t)) = \text{the } k^{th} \text{ component of } (I_N - \beta L(t))\ y$$

where $L(t)$ is the Laplacian of $\mathbb{G}(t)$. With the assumption $\beta \in (0, \frac{1}{d_{max}})$, matrix $(I_N - \beta L(t))$ is doubly stochastic, such that $\|(I_N - \beta L(t))\, y(t)\|^2 \leq \|y(t)\|^2$. This implies

$$W(t+1) - W(t) \leq \|y(t+1)\|^2 - \|(I_N - \beta L(t))\, y(t)\|^2\,. \tag{6.17}$$

Next, define $e^{i\theta}(t) \in \mathbb{C}^N$ to be the complex vector containing all $e^{i\theta_k(t)}$. Moreover, define the particular inner product $\langle y,\, z\rangle$ between $y \in \mathbb{C}^N$ and $z \in \mathbb{C}^N$ to be equal to the scalar product $y^T z$ with y and z considered as elements of $\mathbb{R}^{2N}$; this actually means that $\langle y,\, z\rangle = \Re e\,(y'z)$ where $'$ denotes complex conjugate transpose. Then

$$\begin{aligned}
&\|y(t+1)\|^2 - \|(I_N - \beta L(t))\, y(t)\|^2 \\
&= \langle y(t+1),\ e^{i\theta(t+1)} - e^{i\theta(t)}\rangle + \langle e^{i\theta(t+1)} - e^{i\theta(t)},\ (I_N - \beta L(t))\, y(t)\rangle \\
&= 2\langle y(t+1),\ e^{i\theta(t+1)} - e^{i\theta(t)}\rangle\ -\ \|e^{i\theta(t+1)} - e^{i\theta(t)}\|^2\ \leq\ -\ \|e^{i\theta(t+1)} - e^{i\theta(t)}\|^2
\end{aligned} \tag{6.18}$$

where the last inequality uses the property $\langle y_k(t+1),\ e^{i\theta_k(t+1)}\rangle \leq \langle y_k(t+1),\ e^{i\theta_k(t)}\rangle\ \ \forall k \in \mathcal{V}$ which holds by definition of $\theta_k(t+1)$. Using (6.17) and (6.18) and summing over t yields

$$\sum_{t=0}^{+\infty} \|e^{i\theta(t+1)} - e^{i\theta(t)}\|^2 \leq W(0)\ .$$

The rest of the proof follows lines similar to the continuous-time counterpart (i.e. Proposition 6.2.2). □

Like for the continuous-time algorithm, in practice, the swarm seems to always converge to balanced configurations. ◇

Ex. 6.2.5: consensus with auxiliary variables on $SO(n)$ and $Grass(p,n)$: The particular balancing algorithms are not detailed since they are directly obtained from their synchronization counterparts. Introducing auxiliary matrices $Y_k \in \mathbb{R}^{n\times n}$, (6.8) may be transcribed verbatim. Using previously presented expressions for $\text{Proj}_{T\mathcal{M},k}(Y_k)$, (6.9) becomes

$$\text{On } SO(n): \qquad Q_k^T \tfrac{d}{dt} Q_k = \tfrac{\gamma_S}{2}\left(Q_k^T Y_k - Y_k^T Q_k\right) \quad , \; k = 1, 2, ..., N. \tag{6.19}$$

$$\text{On } Grass(p,n): \qquad \tfrac{d}{dt}\Pi_k = \gamma_S\left(\Pi_k Y_k \Pi_{\perp k} + \Pi_{\perp k} Y_k \Pi_k\right) \; , \; k = 1, 2, ..., N. \tag{6.20}$$

For $Grass(p,n)$, the projector representation *must* be used in (6.8) and (6.11), such that using $n \times n$ matrices becomes unavoidable. ◇

Communication of estimator variables. In order to implement the algorithms of the present section, interconnected agents must communicate the values of their auxiliary variable y_k. The geometric interaction of x_k and y_k implies that they "have the same meaning", i.e. they both represent absolute positions with respect to a hypothetical reference frame in the embedding space $\mathbb{R}^m$ of $\mathcal{M}$. This is in agreement with the invariance analysis made after equations (6.8),(6.9): the same transformation $Q \in \mathcal{G} \subseteq SO(n)$ — corresponding to translation on $\mathcal{M}$, or equivalently rotation of vectors in $\mathbb{R}^m$ — is applied to both the x_k and the y_k. An issue remains about how to communicate the auxiliary variables among agents. Indeed, it is reasonable to assume that relative positions on $\mathcal{M}$ can be measured since the original system lives on $\mathcal{M}$, but the embedding space $\mathbb{R}^m$ is just chosen for mathematical convenience and nothing ensures a priori how agents can exchange the "relative positions" of their auxiliary variables in $\mathbb{R}^m$.

As a first possibility, it is not excluded that the y_k can be represented by physical quantities, in an actually physically meaningful space $\mathbb{R}^m$, such that relative values of the y_k can actually be measured by the agents, as for the x_k. This is however unlikely.

As a second possibility, it may be feasible to use an existing, external reference frame for $\mathbb{R}^m$ in which all the agents can express the coordinates of their y_k. This condition is not difficult to satisfy in computational frameworks, but it may not hold for decentralized control of physical systems: nothing ensures the preliminary existence of a common reference frame on which all agents agree. In addition, relying on a common external reference frame implies that the swarm loses its full autonomy, even if the external frame is just used for "translation" purposes and does not interfer with the dynamics of the system.

As an ideal solution, the agents should express their auxiliary variable y_k in "agent-centered coordinates", relative to their position x_k, as is done on the circle with the change of variables $y_k \to r_k = y_k e^{-i\theta_k}$, see Section 4.3 and Example 6.2.3. The following Example 6.2.6 shows how this can be done for $\mathcal{M} = SO(n)$. It is however unclear how to extend this strategy to other connected compact homogeneous manifolds.

O.Q.: The question of knowing if the auxiliary variables y_k, required by the algorithms of the present section, can be transmitted in "agent-centered" coordinates — i.e. as arrays of scalars without requiring the help of a common external reference frame — on general connected compact homogeneous manifolds, is currently open.

The use of auxiliary variables proposed in the present dissertation may not be optimal in terms of the required information storage and communication. For one thing, although

the embeddings in the examples yield handy expressions for the controllers, they often use variables of higher dimension than needed. For instance, embedding $SO(3)$ in $\mathbb{R}^{3\times 3}$ uses 9-dimensional Y_k; but the 3-dimensional $SO(3)$ can be embedded in $\mathbb{R}^{3\times 2}$ by only retaining the first two columns of Q_k, since the third one is then uniquely determined. It is in fact even possible to embed $SO(3)$ in $\mathbb{R}^5$, see [140].

From a general viewpoint, the idea behind the use of auxiliary variables can be seen as compensating the "missing" information, from the fact that all agents do not communicate with all other agents, by an increase of the information packages exchanged among interconnected agents. In its current form, the algorithm with auxiliary variables requires both increased information storage — meaning, the auxiliary variables use a separate *dynamic* algorithm and thus an additional set of *state variables* in the terminology of state machines — and increased communication — meaning, all components of the auxiliary variable are transmitted among interconnected agents. It may be possible to reduce the amount of additional information stored or communicated. Currently this is another open question.

O.Q.: It is currently an open question whether there is a way to design a synchronization and/or balancing algorithm with similar performance as those proposed in Section 6.2.1, but which makes better use of auxiliary variables: either (i) requiring auxiliary variables of smaller dimension or more to the point, (ii) not requiring explicit *communication* of auxiliary variables among agents, or (iii) requiring communication of additional static information, without a separate dynamic algorithm for the auxiliary variables.

Ex. 6.2.6: communicating auxiliary variables on $SO(n)$: When $\mathcal{M} = SO(n)$, the algorithms of Section 6.2.1 can be reformulated such that they work completely autonomously if interconnected agents measure their relative positions $Q_k^T Q_j$. Indeed, define $Z_k = Q_k^T Y_k$. Then (6.8),(6.9) for instance becomes

$$\dot{Z}_k = (Q_k^T \dot{Q}_k)^T Z_k + \beta \sum_{j=1}^{N} a_{jk} \left((Q_k^T Q_j) Z_j - Z_k \right) \tag{6.21}$$

$$Q_k^T \dot{Q}_k = \tfrac{\gamma_S}{2} \left(Z_k - Z_k^T \right) \quad , \quad k = 1, 2, ..., N \,. \tag{6.22}$$

In this formulation, each agent k represents Z_k as an array of scalars, whose columns express the column-vectors of Y_k as coordinates in a local frame attached to k, i.e. a frame rotated by Q_k with respect to a hypothetical reference frame. Pre-multiplying Z_j by $Q_k^T Q_j$ expresses Y_j in the local frame of k, and $Q_k^T \dot{Q}_k$ expresses the velocity of Q_k (with respect to a hypothetical fixed reference) in the local frame of k as well. Thus (6.21),(6.22) actually correspond to (6.8),(6.9) written in the local frame of k. Each agent k gets from its neighbors $j \rightsquigarrow k$ their relative positions $Q_k^T Q_j$ and the $n \times n$ arrays of numbers Z_j. From this it computes the update $\dot{Z}_k$ to its own array of numbers Z_k and the move it has to make with respect to its current position, $Q_k^T \dot{Q}_k$. The same can be done for the anti-consensus algorithm. ⋄

6.2.2 Modified interaction profile

Section 4.1 proposes a modified attraction profile such that, for any fixed connected undirected graph $\mathbb{G}$, the only stable configuration of the gradient algorithm on the circle is synchronization; in other words, the modified cost function has no other local minima (becoming *maxima* in the present context) than synchronization. The general idea in Section 4.1 goes as follows.

- Local minima of V_θ require agents connected to k to be "sufficiently close" to θ_k.
- Thus, if the interaction profile applied to $(\theta_j - \theta_k)$ is distorted such that the region "sufficiently close" to θ_k becomes small and the region "not sufficiently close" to θ_k is dilated, then the agents connected to k have to be even closer to θ_k.
- Taking the region "sufficiently close" to θ_k small enough, the requirement "θ_j sufficiently close to θ_k" for all edges (j,k) of $\mathbb{G}$ necessarily puts all the θ_k within a semicircle.
- Since, independently of the (reasonably chosen) interaction profile, agents within a semicircle always converge to synchronization by convexity arguments, the only possible minimum of the modified V_θ is synchronization.

The critical part is the first observation: it is not intuitively clear that *all* the in-neighbors of k have to be close to θ_k *for all possible graphs* $\mathbb{G}$ at a local minimum of V_θ. However, Proposition 4.1.2 shows that the idea indeed works.

The general idea can be carried over to connected compact homogeneous manifolds. For simplicity, notation $x_{jk} \in \mathcal{M}$ is used to denote the relative position of x_j with respect to x_k on $\mathcal{M}$. For instance, if $\mathcal{M} = S^1$ then $x_{jk} = e^{i(\theta_j - \theta_k)}$ and if $\mathcal{M} = SO(n)$ then $x_{jk} = Q_k^T Q_j$.

- It seems as reasonable as for the circle that in-neighbors of agent k must be "sufficiently close to" x_k on $\mathcal{M}$ for a local maximum of V_L (i.e. a consensus configuration).
- Then, instead of using the cost function $V_L(\{x_{jk} : j \rightsquigarrow k,\ k \in \mathcal{V}\})$, consider the cost function $V_L(\{\phi(x_{jk}) : j \rightsquigarrow k,\ k \in \mathcal{V}\})$ where ϕ is an automorphism from $\mathcal{M}$ to itself which "distorts" the interaction profile among interconnected agents. The goal of ϕ is to make agents j and k interact as if they were "sufficiently close" for algorithm (6.3) only if they are in fact "very close" to each other, and in the largely dominant remaining part of $\mathcal{M}$ make them interact as if they were "not sufficiently close" for algorithm (6.3).
- With the profile suitably modified by ϕ, a configuration with agents spread on $\mathcal{M}$ is necessarily an unstable point of the gradient algorithm; at stable configurations, agents would have to be so close that they are necessarily within a convex set of $\mathcal{M}$.
- Finally, similarly to the case of S^1, by convexity the only maximum of $V_L(\{\phi(x_{jk}) : j \rightsquigarrow k,\ k \in \mathcal{V}\})$ would be synchronization.

Currently, there are two limitations to turn these ideas into a formal result. First, a general method has to be proposed to design the automorphism $\phi : \mathcal{M} \to \mathcal{M}$ in a way that is both geometrically acceptable (at least, the resulting gradient algorithm should be continuous) and "sufficiently tightening" for local maxima; a quantitative formulation of the second requirement depends on the characterization of the largest possible convex set on $\mathcal{M}$. Secondly, and most importantly, a formal proof is necessary to confirm the first intuitive observation and thus the synchronization property of the algorithm. The proof in Section 4.1 is algebraic rather than geometric, and greatly simplified by the fact that S^1 is one-dimensional. This proof has not been extended to connected compact homogeneous manifolds, even though no particular obstacle is expected.

O.Q.: The design of a "modified profile" algorithm for synchronization on general connected compact homogeneous manifolds, along with a formal convergence proof, is currently an open question.

6.2.3 Introducing randomness in link selection

The basic ingredient in the algorithms of Section 4.2 is that, instead of systematically using information from all agents $j \rightsquigarrow k$, at each time step agent k randomly selects zero or one of its in-neighbors according to a uniform probability distribution. The idea behind this procedure is that there is a finite probability that the union of the agents' edge choices will form a tree-like graph for a long enough time such that the agents are driven into a semicircle; once this is achieved, by convexity arguments, any chosen link can only enhance convergence towards synchronization. Section 4.2 examines two settings, corresponding to "directed gossip edges" — mimicking behavior of directed graphs — and "undirected gossip edges" — mimicking behavior of undirected graphs. The following discusses their extension to connected compact homogeneous manifolds; similarly to Section 4.2, a discrete-time framework is adopted for simplicity of formulation.

In the setting of directed gossip, the algorithm proposed in Section 4.2 is applicable to *any* manifold (in fact any state space, even discrete). It is rewritten here for convenience.

Gossip algorithm (directed). At each update t,

1. each agent k randomly selects an agent $j \rightsquigarrow k$ with probability $a_{jk}/\left(\beta + \sum_{l \rightsquigarrow k} a_{lk}\right)$, where $\beta > 0$ is the weight for choosing no agent;
2. $x_k(t+1) = x_j(t)$ if agent k chooses neighbor j at time t, and $x_k(t+1) = x_k(t)$ if it chooses no neighbor.

Moreover, the argument used in the proof of convergence for this algorithm is not specific to the circle. Therefore, Proposition 4.2.4 actually holds on any manifold, as restated below; see Definition 4.2.1 of "asymptotic synchronization with probability 1".

Proposition 6.2.7: *Consider a set of N agents on any state space $\mathcal{M}$, interconnected according to a uniformly connected δ-digraph $\mathbb{G}$ and applying the directed Gossip Algorithm, with a fixed $\beta > 0$. Then the swarm asymptotically synchronizes with probability 1.*

Proof: The formal proof is the same as for Proposition 4.2.4 on the circle, modulo trivial adaptations of notation. Instead of repeating it, an intuitive summary is given here. The idea is that, over a sufficiently long but finite time span T_s, there is a finite probability p_s to observe a sequence where the union of chosen links forms a directed tree *and* the agents consecutively move towards their parents in such an order that they actually all jump to the position of the root of the tree. This would imply synchronization from any initial condition. Therefore, there is a finite probability $\geq p_s$ that the agents synchronize during any time span T_s. As a consequence, the probability that the agents do not synchronize decreases at least as $(1-p_s)^{t/T_s}$. □

The algorithm proposed in Section 4.2 in the setting of undirected gossip can be generalized as follows to connected compact homogeneous manifolds.

Gossip algorithm (undirected). At each update t,

1. each agent k randomly selects one neighbor or none, as in the directed version;
2. if k chooses j `AND` j chooses k at time t, then k and j move to their induced arithmetic mean, i.e. $x_k(t+1) = x_j(t+1) = IAM(x_k(t),\, x_j(t))$; if the *IAM* contains several points, then both agents move to a point $\in IAM(x_k(t),\ x_j(t))$ that is equidistant from $x_k(t)$

and $x_j(t)$ — the same point for j and k. If k chooses no neighbor or a neighbor j which does not choose k, then $x_k(t+1) = x_k(t)$.

For the undirected Gossip Algorithm, the convergence proof on the circle is based on the existence of a "synchronizing sequence" which appears with finite probability in some bounded time span and after which all agents end up in a semicircle, for all initial positions. The synchronizing sequence in Proposition 4.2.5 basically holds because the largest convex set of S^1 is an open semicircle, such that when an agent moves to the *IAM* of itself and its bilaterally chosen neighbor, it maximally travels by an arclength of $\pi/2$ *which does not exceed the arclength radius of a convex set.* Nothing seems to a priori ensure a similar property for general compact connected homogeneous manifolds; it is therefore not clear if and how the synchronizing sequence used in the proof of Proposition 4.2.5 can be adapted. The arclength property does hold for the sphere of arbitray dimension S^n, as well as for $SO(n)$ as confirmed by [52]. The following extends Proposition 4.2.5 to the sphere S^n.

Proposition 6.2.8: *Consider a set of N agents on the sphere S^n, interconnected according to a uniformly connected undirected graph $\mathbb{G}$ and applying the undirected Gossip Algorithm, with a fixed $\beta > 0$. Then the swarm asymptotically synchronizes with probability 1.*

Proof: The argument is the same as for Proposition 4.2.5, using the exact same synchronizing sequence. The convex set "semicircle" is replaced by the "semisphere", which is an arbitrary rotation of the restriction of S^n to the elements of $\mathbb{R}^{n+1}$ whose first component is positive.

A sequence as used in the proof of Proposition 4.2.5 is still synchronizing on S^n. Indeed, consider that a set $\mathcal{S}_M$ of $M < N$ agents is located within a "$2^{(n+1)}$-ant" of the sphere, i.e. an arbitrary rotation of the restriction of S^n to the elements of $\mathbb{R}^{n+1}$ whose components are *all* positive; this set has geodesic diameter $\frac{\pi}{2}$. Now at time t let one agent j of this set interact with a new agent k to be "added" to $\mathcal{S}_M$ in order to build $\mathcal{S}_{M+1}$. There is always a great circle containing positions $x_j(t)$ and $x_k(t)$, and the new position $x_j(t+1) = x_k(t+1)$ is located on the same great circle, at a geodesic distance at most $\frac{\pi}{2}$ from $x_j(t)$. Therefore, the new positions of the M agents in $\mathcal{S}_M$ and of the new agent k are located within a set of diameter less than π. After this, by convexity, applying sufficiently many interactions among the agents of $\mathcal{S}_M \cup \{k\}$, while avoiding interactions with other agents, allows to shrink the minimal set containing $\{x_j : j \in \mathcal{S}_M \cup \{k\}\}$ to an arbitrarily small size. Once it is contained in a "$2^{(n+1)}$-ant", the argument can be repeated recursively by adding a new agent to $\mathcal{S}_{M+1} := \mathcal{S}_M \cup \{k\}$. The remainder of the argument is exactly as in the proof of Proposition 4.2.5. The only difference might be in the number of interactions that is required among elements of $\{x_j : j \in \mathcal{S}_M \cup \{k\}\}$ until the convex set obtained after the first interaction with k has sufficiently shrinked. This is unimportant for the convergence argument as long as the number stays bounded. □

O.Q.: It is currently an open question if asymptotic synchronization with probability 1 can be proved for the undirected Gossip Algorithm on general connected compact homogeneous manifolds.

Chapter 7

Attitude synchronization of mechanical systems

Most results of the present disseration consider velocity-controlled first-order integrator agents. The present chapter illustrates how the proposed tools and algorithms can be used in more complex and realistic settings. For this purpose, *synchronization* is chosen as an example task, and the control of 3-dimensional orientations — also often called *attitudes* — of free rigid bodies obeying the laws of Newtonian mechanics, with torques as control inputs, is chosen as an example system. Beyond its theoretical importance as the prototype for nonlinear control of mechanical systems on Lie groups, rigid body attitude synchronization is of practical interest for instance in the framework of space missions involving multiple satellites; see Chapter 1 and the Introduction to Part II for more details.

Of course, mechanical dynamics can also be considered in the framework of consensus algorithms on *vector spaces*. This has been done in recent literature. Position synchronization on a vector space is easier than attitude synchronization in two respects. First, the agreement problem is easier in a vector space, as is abundantly discussed throughout this dissertation. Second, the Newtonian dynamics for position control of point-mass particles are linear, whereas the Euler equations for the control of rigid body orientations are fundamentally nonlinear.

About quaternions: In the present dissertation, a 3-dimensional orientation is represented by a rotation matrix. In the literature and in practical implementations, orientations are often parametrized by e.g. the Euler angles or, most popularly, the quaternions of unit norm. A short discussion of at least the latter is thus appropriate at this point.

Generally speaking, the set of *quaternions* $\mathbb{Q}$ is just equivalent to $\mathbb{R}^4$; however, in addition to scalar multiplication and vector addition, a multiplication operation is defined among quaternions, i.e. such that the product of two quaternions is a quaternion, see [47]. This is similar to the multiplication which is defined on $\mathbb{R}^2$ by identifying it with the complex numbers $\mathbb{C}$, except that quaternion multiplication is non-commutative, just like group multiplication (but in contrast to group multiplication, it does not admit an inverse for each quaternion).

The set of *unit quaternions* $\{q \in \mathbb{Q} : \|q\| = 1\}$, with $\| \ \|$ the Euclidean norm of $\mathbb{R}^4$, is equal to the sphere S^3. The quaternion multiplication defines a Lie group structure on S^3. This group forms a *double cover* of $SO(3)$ which is used as follows to parametrize 3-dimensional

rotations. For a rotation of angle θ around normalized axis $\mathbf{e}$, the first three components (also called the vector component) of the corresponding quaternion are set to $\sin(\frac{\theta}{2})\,\mathbf{e}$ and the last component (also called the scalar component) is set to $\cos(\frac{\theta}{2})$. The advantage of the quaternion parametrization is that it uses only 4 numbers to characterize a rotation — less than the 9 components of a rotation matrix or the 5 components of a minimal embedding of $SO(3)$ (see [140]) — and unlike parametrizations with Euler angles or its derivatives, it has no singularity: for any rotation, even the identity, there are exactly two unit quaternions q and $-q$. This makes the quaternions a very interesting parametrization of rotations in applications.

Regarding practical implementation, nothing prevents to use quaternions to encode the algorithms of the present dissertation concerning $SO(3)$, using an appropriate treatment of the double cover $Q \in SO(3) \leftrightarrow \{q, -q\} \in \mathbb{Q}$. However, the existence of *two* quaternions for one rotation clearly shows that the global structure of $SO(3)$ is different from $S^3 \subset \mathbb{Q} \cong \mathbb{R}^4$. Therefore one must be careful when designing algorithms, where the use of quaternions could lead to unphysical behavior. For instance, if two agents k and j have nearly the same orientation but with $q_k \approx -q_j$, then a synchronization algorithm on quaternions would move the orientations apart, to reach $q_k = q_j$ by synchronizing at a possibly completely different place of $SO(3)$. A common convention in the literature to avoid such unphysical behavior is to allow only unit quaternions with positive scalar part, in order to have one quaternion for each orientation. This amounts to mapping $SO(3)$ to the semisphere S^3 and unavoidably introduces a discontinuity at the edge of the semisphere, which corresponds to rotations of 180 degrees. This discontinuity is not harmful when one studies local behavior — unlike for the Euler angles, it is not in the neighborhood of identity — and can be easily treated in more general situations when an external reference fixes a place of discontinuity for *absolute* orientations *in inertial frame*, common to all the agents. However when the agents use their own local frame in a distributed autonomous-agent setting and are distributed on all $SO(3)$, it becomes more difficult to handle the resulting multiple discontinuities or the double representation by q and $-q$.

To avoid these difficulties, since a main focus of the present dissertation is the global geometry of the configuration space, all reasoning and algorithms on $SO(3)$ are developed with rotation matrices. The algorithms may then be *translated* into quaternions or other parametrizations for practical implementation. It should be clear that difficulties related to the fundamental geometric properties of $SO(3)$ cannot be solved simply by switching to a quaternion or other parametrization without fundamentally altering the physical problem.

7.1 Synchronization of mechanical systems

7.1.1 Mechanical systems on vector spaces

As recalled in Section 3.1, synchronization on vector spaces under first-order integrator dynamics is a well-studied problem in the literature. Numerous authors have also considered synchronization of agents with mechanical system dynamics, or even more general dynamics, whose configuration space is a vector space. A review of corresponding results goes beyond the scope of the present dissertation; relevant work is cited along the discussion. Consider a swarm of N identical particles whose positions on a vector space $\mathbb{R}^m$ follow Newton's law

$$\frac{d^2}{dt^2} x_k = u_k \quad , \quad k = 1, 2, ..., N \,. \tag{7.1}$$

where u_k is the force applied to particle k. The agents are assumed to be *isolated*, i.e. the force u_k contains nothing else than the control input of agent k. For $u_k = 0$, Newton's law, like all known physical laws, is invariant with respect to absolute position, i.e. adding a fixed position-vector a to all agents $(x_1, x_2, ..., x_N) \to (x_1 + a, x_2 + a, ..., x_N + a)$ does not change the dynamics for the evolution of relative positions in the swarm. In order to maintain this symmetry, a nonzero u_k should only depend on relative positions $x_j - x_k$. Newton's law (7.1) with $u_k = 0$ is also invariant with respect to absolute velocity, i.e. adding a fixed velocity-vector b to all agents $(x_1, x_2, ..., x_N) \to (x_1 + bt, x_2 + bt, ..., x_N + bt)$, or equivalently $(v_1, v_2, ..., v_N) \to (v_1 + b, v_2 + b, ..., v_N + b)$ after defining $v_k := \frac{d}{dt}x_k$, does not change the dynamics for the evolution of relative positions in the swarm. In order to maintain this symmetry, a nonzero control u_k should only depend on relative velocities $v_j - v_k$.

Results for synchronization in vector spaces in the setting (7.1), where u_k depends on relative positions and relative velocities of connected agents $j \rightsquigarrow k$, can be found among others in [73, 112, 125] and references therein. They are included here for comparison with the nonlinear problem of attitude synchronization. The interconnection graph $\mathbb{G}$ is assumed to be unweighted. The classical controller for synchronizing agent positions in the setting (7.1) is

$$u_k = \alpha_s \sum_{j \rightsquigarrow k} (x_j - x_k) + \alpha_d \sum_{j \rightsquigarrow k} (\tfrac{d}{dt}x_j - \tfrac{d}{dt}x_k) \quad , \quad k = 1, 2, ..., N \,, \tag{7.2}$$

with $\alpha_s > 0$ and $\alpha_d > 0$. For fixed x_j, (7.2) assigns to agent k the dynamics of a damped spring, with spring rigidity proportional to α_s and damping coefficient proportional to α_d. This is a natural generalization from linear first-order to linear second-order asymptotically stable systems. It can also be interpreted considering passivity properties or energy shaping (for fixed undirected graphs). The latter technique is used in Section 7.3 to design controllers for attitude synchronization and the reader is referred to that section for further explanation. The convergence properties of (7.1),(7.2) are *not* as straightforward as Proposition 3.1.1 for consensus in vector spaces with first-order integrator dynamics. Results in [73, 112] can be summarized as follows and show that some basic tuning is necessary to reach synchronization on vector spaces in a mechanical setting.

Proposition 7.1.1: *(adapted from [112]) Consider a swarm of N agents evolving on a vector space $\mathbb{R}^m$ according to algorithm (7.1),(7.2) with $\alpha_s > 0$ and $\alpha_d > 0$, with communication links among agents characterized by the edges of a graph $\mathbb{G}$.*

(a) *If $\mathbb{G}$ is fixed, undirected and connected, then for all initial conditions $(x_k(0), v_k(0))$, $k = 1, 2, ..., N$, the agents exponentially synchronize.*

(b) *If $\mathbb{G}$ is fixed, directed and strongly connected, then the agents exponentially synchronize for all initial conditions $(x_k(0), v_k(0))$ if α_d is sufficiently large (with respect to α_s).*

(c) *If $\mathbb{G}$ is time-varying, directed and strongly connected at all times, then the agents asymptotically synchronize for all initial conditions $(x_k(0), v_k(0))$ if α_d is sufficiently large and the graph $\mathbb{G}$ only switches at time instants that are sufficiently far apart.*

Proof: See reference [112]. □

The term multiplied by α_d is usually called the *dissipation term*. It decreases the "energy" of the system such that the latter is not only stable, but also *asymptotically* stable as required for convergence towards synchronization; more explanation is given in Section 7.3 where this concept is taken over for attitude synchronization. Proposition 7.1.1 thus essentially says

that convergence towards synchronization is ensured if the dissipation term is large enough. This is in fact not too surprising, since for $\alpha_d \gg \alpha_s$, algorithm (7.1),(7.2) becomes close to a single-integrator consensus algorithm for velocities in $\mathbb{R}^m$,

$$\frac{d}{dt} v_k = \alpha_d \sum_{j \rightsquigarrow k} (v_j - v_k) \quad , \quad k = 1, 2, ..., N \, ,$$

ensuring at least synchronization of velocities v_k.

Interestingly, using dynamic controllers with auxiliary variables allows to recover the synchronization properties of the single integrator (Proposition 3.1.1), with no additional conditions, for double integrator dynamics on vector spaces. This is shown in [125] for the more general case of linear systems. Thus the exchange of auxiliary variables can be advantageous not only for synchronization on *manifolds*, as in Section 6.2.1, but also for synchronization of *general linear systems* in vector spaces. The link between manifolds and linear systems for the role of auxiliary variables is still under investigation; one idea might be to look at the example of harmonic oscillator dynamics: although the state space of a single oscillator is the plane, its trajectories are circular.

7.1.2 Mechanical systems on $SO(3)$

The mechanical model for the evolution of rigid body orientations can be computed as a generalization from Newton's equation of motion for particles (7.1); see [34] for instance. This leads to the so-called Euler equations of rigid body rotation. Representing rigid body orientations by rotation matrices $Q_k \in SO(3)$ as in Chapters 5 and 6, a kinematic equation describes the evolution of orientation Q_k as a function of the *angular velocity* ω_k by

$$\frac{d}{dt} Q_k = Q_k [\omega_k]^\wedge \quad , \quad k = 1, 2, ..., N \, . \tag{7.3}$$

The invertible operator $[\cdot]^\wedge$ defined by

$$a = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} \in \mathbb{R}^3 \; \leftrightarrow \; [a]^\wedge = \begin{pmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{pmatrix} \in \mathfrak{so}(3)$$

identifies the tangent space to $SO(3)$ at identity I_3, which is the set $\mathfrak{so}(3)$ of antisymmetric matrices, with $\mathbb{R}^3$. Denoting by $\wedge$ the vector product, $[a]^\wedge b = a \wedge b$, $\forall a, b \in \mathbb{R}^3$. The inverse of $[\cdot]^\wedge$ is denoted $[\cdot]^\vee : \mathfrak{so}(3) \to \mathbb{R}^3$. The advantage of using this notation is that ω_k has the physical interpretation of being the *angular velocity* of body k with respect to an arbitrary inertial frame, *expressed in a body frame* attached to agent k. The evolution of angular velocity as a function of *torques* applied to the body is given by

$$J_k \frac{d}{dt} \omega_k = (J_k \omega_k) \wedge \omega_k + u_k \quad , \quad k = 1, 2, ..., N \, , \tag{7.4}$$

where $J_k = \mathrm{diag}(J_{k1}, J_{k2}, J_{k3}) \in \mathbb{R}^{3\times 3}$ is the moment of inertia matrix of agent k and $u_k \in \mathbb{R}^3$ its control torque; both quantities are expressed in body frame and $J_{k1} \geq J_{k2} \geq J_{k3}$ without loss of generality. The torques and velocities expressed in inertial frame are then $Q_k u_k$ and $Q_k \omega_k$ respectively. Considering different moments of inertia J_k for the different agents is a departure from the basic assumption of identical agents; this is just made to highlight where identical J_k are a necessary condition for proper convergence of the mechanical algorithms.

Symmetries and invariance: Similarly to vector spaces, for isolated agents $u_k = 0$, dynamics (7.3),(7.4) are invariant with respect to a fixed rotation of all rigid bodies: the dynamics for the evolution of relative orientations in the swarm do not change under the transformation $(Q_1, Q_2, ..., Q_N) \to (QQ_1, QQ_2, ..., QQ_N)$ with $Q \in SO(3)$. In order to maintain this invariance, the control torques u_k may only depend on *relative* orientations $Q_k^T Q_j$ and not on *absolute* orientations Q_k.

For velocities however, the case is different from vector spaces. Indeed, even for $u_k = 0$, equation (7.4) depends — nonlinearly — on ω_k. This is a direct consequence of the Lie group structure of $SO(3)$ (see e.g. [5]), the vector product $\wedge$ being the Lie bracket associated to $\mathfrak{so}(3) \cong \mathbb{R}^3$. On vector spaces, the Lie bracket trivially vanishes. The fact that natural mechanical equations of motion on nonlinear manifolds can be invariant with respect to configuration symmetries but *irreducibly* depend on the associated velocities — i.e. with no external influence (force or torque), velocities cannot be factored out of the dynamics of the relative configuration — is well-known in the study of mechanical systems. *Reduction techniques* study system properties related to this fact; see [82, 83], and [48, 135] for a specific discussion on Lie groups $SO(3)$ and $SE(3)$.

For the agreement problem considered in the present chapter, the presence of ω_k in the free rigid body dynamics implies that *absolute angular velocities* ω_k and $\{\omega_j : j \rightsquigarrow k\}$ can a priori be used in the control input u_k without breaking the natural symmetry of the system, since the latter is *not* invariant with respect to the transformation $(Q_1\omega_1, Q_2\omega_2, ..., Q_N\omega_N) \to (Q_1\omega_1 + b, Q_2\omega_2 + b, ..., Q_N\omega_N + b)$ assigning the same fixed rotational motion to all agents of the swarm. In fact, the dependence of free rigid body dynamics on ω_k implies that inertial sensor devices can actually measure ω_k without needing any external reference; thus using ω_k in the control inputs is not bad in terms of autonomous operation of the agents. A more stringent requirement would be to design input torques on the basis of *relative angular velocities* $Q_k\omega_k - Q_j\omega_j$ only; then u_k can only depend on $\{Q_k^T Q_j : j \rightsquigarrow k\}$ and $\{Q_k^T Q_j \omega_j - \omega_k : j \rightsquigarrow k\}$. This stronger requirement "keeps the free rigid body dynamics", such that the motion of a synchronized swarm can be any solution of (7.4) with $u_k = 0$. In contrast, controllers using absolute angular velocities generally "destroy" the free rigid body dynamics, reducing (or altering) the set of possible asymptotic motions of a synchronized swarm.

Rigid body attitude synchronization is more difficult than point-mass position synchronization in two respects.

- The agreement problem is fundamentally more difficult on nonlinear manifold $SO(3)$ than on a vector space; this is a fundamental issue of the present dissertation.
- In addition, independently of the agreement problem, the manifold structure implies that the mechanical dynamics of a freely moving object on $SO(3)$ are nonlinear, thereby much more complex than the corresponding dynamics on a vector space (double integrator); this is an issue for any control problem on nonlinear manifolds.

The following sections design control algorithms for (7.3),(7.4) with control u_k allowed to depend on relative orientations $Q_k^T Q_j$ and absolute angular velocities ω_k. A restrictive simplification in this setting is that the agents are assumed fully actuated, i.e. the range of u_k is $\mathbb{R}^3$; an extension to underactuated agents is however not too difficult to imagine. Two approaches are considered to design the input torques u_k. The first method, called *consensus tracking*, takes a backstepping approach; the second method, *energy shaping*, follows another tradition for control of mechanical systems. The basic ideas of each approach are

described at the beginning of the respective sections. With the energy shaping approach, a local synchronization result for fixed undirected graphs but using only *relative* angular velocities ($Q_k^T Q_j \omega_j - \omega_k$) is also developed; it requires the controller gain to dominate the free rigid body dynamics in the Euler equation (7.4). The advantage of using relative velocities is that the resulting controller allows to asymptotically sychronize attitudes on an unspecified trajectory that can be *any free rigid body motion*, thus somehow "maintaining symmetry with respect to the natural system dynamics".

For the sake of completeness, it may be noted that observers can retrieve the angular velocities ω_k using measures of orientations with respect to an inertial reference frame (among others see [15] and so-called "passivity filters"). This would imply that the swarm, relying on a common external reference, looses its full autonomy. However, the external frame is just used for "observation" purposes and does not interfer with the dynamics of the system; invariance with respect to a fixed rotation of all the agents can be maintained.

7.2 Consensus tracking

The idea of what is here called "consensus tracking" follows the *backstepping* approach, a classical method for controller design. In a first step, the rigid bodies are considered to be velocity-controlled, i.e. modeled by (7.3) with ω_k as control inputs, and a suitable feedback control $\omega_k^{(d)}$ is designed to reach synchronization. Then a second step considers the $\omega_k^{(d)}$ as reference velocities ("desired velocities") to track with the complete, mechanical model (7.3),(7.4) controlled through torques u_k.

The first step exactly corresponds to the setting of Chapter 6. The algorithms designed in that chapter are summarized as follows.

- **A gradient algorithm** proposed in Section 6.1,

$$\omega_k^{(d)} = \alpha \sum_{j=1}^{N} a_{jk} \left[Q_k^T Q_j - Q_j^T Q_k\right]^\vee , \quad k = 1, 2, ..., N \text{ and } \alpha > 0 , \tag{7.5}$$

  asymptotically stabilizes consensus configurations for fixed undirected $\mathbb{G}$; for a tree or complete graph, synchronization is the only stable equilibrium.

- **An algorithm with auxiliary variables** $Z_k \in \mathbb{R}^{3\times 3}$ proposed in Section 6.2.1 writes

$$\begin{aligned} \tfrac{d}{dt} Z_k &= \beta \sum_{j=1}^{N} a_{jk}((Q_k^T Q_j) Z_j - Z_k) \; - [\omega_k]^\wedge Z_k & (7.6)\\ \omega_k^{(d)} &= \tfrac{\gamma_S}{2} \left[Z_k - Z_k^T\right]^\vee \qquad , \quad k = 1, 2, ..., N , & (7.7) \end{aligned}$$

  with $\beta > 0$ and $\gamma_S > 0$; equation (7.6) uses the *actual* angular velocity, not the desired one: it takes into account the actual motion of a reference frame attached to the body Q_k, according to the change of variables explained at the end of Section 6.2.1. For $\mathbb{G}$ uniformly connected, the only stable behavior of (7.6),(7.7) is synchronization of all Q_k.

The second step requires to design a tracking algorithm making ω_k converge to $\omega_k^{(d)}$ for $k = 1, 2, ..., N$. In this step, the individual agents are decoupled. Attitude tracking algorithms abound in the literature, see among others [15, 22, 46, 70, 80, 85]. The following applies two classical approaches of control theory to tracking of $\omega_k^{(d)}$: *computed torque* and *high gain*. The starting point for both methods is to impose an exponential convergence of ω_k to $\omega_k^{(d)}$:

$$\frac{d}{dt}(\omega_k - \omega_k^{(d)}) = -\mu(\omega_k - \omega_k^{(d)}) \quad , \quad \mu > 0 \; . \tag{7.8}$$

Instead of a scalar, μ can be replaced by any positive-definite matrix; the choice

$$\frac{d}{dt}(\omega_k - \omega_k^{(d)}) = -\mu \, J_k^{-1} \, (\omega_k - \omega_k^{(d)}) \quad , \quad \mu > 0 \tag{7.9}$$

leads to simpler expressions in the present setting. A control torque achieving (7.9) leads to zero final velocity. A fixed nonzero velocity ω_0 in body frame can be imposed by taking desired velocity $\omega_k^{(d)} + \omega_0$, which leads to

$$\frac{d}{dt}(\omega_k - \omega_k^{(d)}) = -\mu \, J_k^{-1} \, (\omega_k - \omega_k^{(d)} - \omega_0) \quad , \quad \mu > 0 \; . \tag{7.10}$$

This also requires to modify (7.6),(7.7), because (7.6) makes the $Y_k := Q_k Z_k$ converge to a *fixed* $\bar{Y}$ and (7.7) drives the agents towards that fixed $\bar{Y}$, which is not compatible with a rotation at velocity ω_0. To obtain a final rotation at ω_0, the agents must track $Y_k e^{t[\omega_0]^\wedge}$ instead of Y_k in (7.7), where e^A with $A \in \mathbb{R}^{3\times 3}$ denotes the canonical matrix exponential (Taylor series definition). Defining $F_k = Z_k e^{t[\omega_0]^\wedge}$, this is equivalent to replacing (7.6),(7.7) by

$$\frac{d}{dt}F_k = \beta \sum_{j=1}^{N} a_{jk}((Q_k^T Q_j)F_j - F_k) \; - [\omega_k]^\wedge F_k + F_k[\omega_0]^\wedge \tag{7.11}$$

$$\omega_k^{(d)} = \frac{\gamma_S}{2} \, [F_k - F_k^T]^\vee \qquad , \quad k = 1, 2, ..., N \; . \tag{7.12}$$

The "computed torque" method simply implements the closed-loop dynamics (7.8) with a control torque that exactly cancels the free rigid body dynamics; this yields

$$u_k = -(J_k \omega_k) \wedge \omega_k + J_k \tfrac{d}{dt}\omega_k^{(d)} + \mu \, (\omega_k^{(d)} + \omega_0 - \omega_k) \quad , \quad k = 1, 2, ..., N \; . \tag{7.13}$$

The main point in (7.13) is to cancel the natural behavior of the system, such that it can be controlled like a linear integrator. This extreme case of feedback linearization is convenient from an analysis viewpoint but raises several performance and robustness issues. These practical issues related to "computed torque" have been addressed in the literature for many other control problems, see e.g. [134].

An alternative to exactly cancelling the natural dynamics, as is done in the computed torque method, is to dominate them by using a sufficiently high gain on the part of the controller pulling the velocity towards its desired behavior. This leaves

$$u_k = J_k \tfrac{d}{dt}\omega_k^{(d)} + \mu \, (\omega_k^{(d)} + \omega_0 - \omega_k) \quad , \quad k = 1, 2, ..., N \; , \tag{7.14}$$

with an additional condition $\mu > \mu^*$. The "high gain" controller works reasonably well without requiring additional adaptation schemes and is therefore more robust than the simple

computed torque controller. Torques resulting from (7.14) can however be large since μ must take a potentially large minimal value; this can cause practical problems for power consumption and actuator saturation.

Several comments are in order about controllers (7.13) and (7.14) before turning to their convergence properties.

- The knowledge of absolute angular velocity ω_k is necessary to implement not only (7.13) but also (7.14). Indeed, implementing (7.14) with only relative velocities would require $\omega_k^{(d)}$ to be a combination of other agents' velocities $Q_k^T Q_j \omega_j$ in the frame of agent k — which however would mean not taking *attitudes* into account, so making it impossible to synchronize them; in particular, see $\omega_k^{(d)}$ in (7.5) or (7.11),(7.12). As a consequence, it is unavoidable that the angular velocity of the synchronized swarm is directly imposed by the controller.
- When the auxiliary variable method is used in the consensus step, a simpler control strategy can be used by defining a desired *orientation* $Q_k^{(d)}$ as the projection of $Y_k = Q_k Z_k$ onto $SO(3)$. Since the auxiliary variables Y_k converge towards a desired value $\bar{Y}$ independently of the actual agents' motion, it may even be useless to move the rigid bodies before the Y_k reach consensus. Subsequently moving the agents individually to $\bar{Q}^{(d)}$ just requires (for $\omega_0 = 0$) global attitude *stabilization* instead of tracking capabilities.
- The consensus tracking approach can also be applied to particles moving in a vector space according to Newton's law. Without using auxiliary variables, this leads to

$$\tfrac{d}{dt} v_k = -\mu\,(v_k - v_k^{(d)}) + \tfrac{d}{dt} v_k^{(d)} = \mu \sum_{j \rightsquigarrow k} (x_j - x_k) + \sum_{j \rightsquigarrow k} (v_j - v_k) - \mu\, v_k \tag{7.15}$$

  which differs from (7.2) by an additional term involving the absolute linear velocity v_k; this would break the symmetry of (7.1) with respect to absolute velocity. A strategy using auxiliary variables is proposed in [125].
- The author of [109] proposes to add $-(J_k\omega_k) \wedge \omega_k$ to a simple control torque derived from energy shaping (see 7.3) in order to get rid of the annoying nonlinear free rigid body dynamics. The remaining part of the local attitude synchronization controller is equivalent to (7.2),

$$u_k = -(J_k\omega_k) \wedge \omega_k + \alpha_s \sum_{j \rightsquigarrow k} [Q_k^T Q_j - Q_j^T Q_k]^\vee + \alpha_d \sum_{j \rightsquigarrow k} (Q_k^T Q_j \omega_j - \omega_k)\,.$$

  Except that initial conditions must be restricted to a neighborhood of synchronization, convergence properties are as for vector spaces in Proposition 7.1.1. This is not surprising since, after applying torque $-(J_k\omega_k)\wedge\omega_k$, the dynamics of Q_k are locally essentially equivalent to the double integrator of (7.1).

Proposition 7.2.1: *Consider a swarm of N rigid bodies communicating along the edges of a δ-digraph $\mathbb{G}$ and whose orientation dynamics follow (7.3),(7.4) with control torque (7.13).*

(a) *If $\mathbb{G}$ is fixed and undirected and $\omega_k^{(d)}$ is defined by (7.5), then the rigid bodies converge to a set where $\omega_k = \omega_0\ \forall k \in \mathcal{V}$ and the orientations are at a critical point of V_L defined in Section 5.3. The only stable solutions regarding relative orientations are consensus configurations of* $\mathbb{G}$ that are compatible with $\omega_k = \omega_0\ \forall k \in \mathcal{V}$, *and if $\mathbb{G}$ is the complete graph or a tree, then the only stable solution is attitude synchronization.*

(b) *If $\mathbb{G}$ is piecewise continuous and uniformly connected, and $\omega_k^{(d)}$ is defined by (7.11),(7.12), then for almost all initial conditions $Z_k(0)$, $Q_k(0)$, $\omega_k(0)$, the solutions converge to attitude synchronization with $\omega_k = \omega_0$ $\forall k \in \mathcal{V}$.*

Proof: Define $\omega_k^{(r)} = \omega_k - \omega_0$. The proof of (a) uses Lyapunov function

$$\begin{aligned} W &= -V_L + \tfrac{\kappa}{2}\sum_{k=1}^{N}(\omega_k^{(r)} - \omega_k^{(d)})^T J_k(\omega_k^{(r)} - \omega_k^{(d)}) \\ &= \tfrac{-1}{2N^2}\sum_{k=1}^{N}\sum_{j=1}^{N} a_{jk}\, Q_k^T Q_j \;+\; \tfrac{\kappa}{2}\sum_{k=1}^{N}(\omega_k^{(r)} - \omega_k^{(d)})^T J_k(\omega_k^{(r)} - \omega_k^{(d)}) \end{aligned} \tag{7.16}$$

where κ is a positive constant to be determined. From the developments of Chapter 6,

$$\tfrac{d}{dt}V_L = \tfrac{1}{\alpha}\textstyle\sum_k \mathrm{trace}\left(-[\omega_k^{(d)}]^\wedge[\omega_k]^\wedge\right) = \tfrac{2}{\alpha}\textstyle\sum_k (\omega_k^{(d)})^T\omega_k$$

because $\mathrm{trace}(-[x]^\wedge[y]^\wedge) = 2x^Ty \quad \forall x, y \in \mathbb{R}^3$. The time-derivative of the remaining term in W is given by the control law. Then a few basic calculations lead to

$$\tfrac{d}{dt}W = \sum_{k=1}^{N} -(\mu\kappa - \tfrac{1}{\alpha})\|\omega_k^{(r)} - \omega_k^{(d)}\|^2 - \tfrac{1}{\alpha}(\|\omega_k^{(r)}\|^2 + \|\omega_k^{(d)}\|^2) - \tfrac{2}{\alpha}(\omega_k^{(d)})^T\omega_0\,. \tag{7.17}$$

One checks from (7.5) that $\sum_k(\omega_k^{(d)}) = 0$, such that the last term vanishes. As a consequence, $\frac{d}{dt}W \leq 0$ by choosing $\kappa > \frac{1}{\mu\alpha}$. Then the LaSalle invariance principle implies that the system converges to the set where $\frac{d}{dt}W = 0$. The latter requires $\omega_k^{(r)} = \omega_k^{(d)} = 0$ $\forall k \in \mathcal{V}$. Condition $\omega_k^{(d)} = 0$ characterizes the critical points of V_L, and $\omega_k^{(r)} = 0$ imposes at the same time $\omega_k = \omega_0$. The stable critical points must be minima of W, hence maxima of V_L; this concludes the proof of (a).

For (b), the $Y_k = Q_kZ_k = Q_kF_ke^{-t[\omega_0]^\wedge}$ follow a linear vector space consensus algorithm, independently of the agents' motion. Therefore, Proposition 3.1.1 ensures that they converge to some common $\bar{Y}$ if $\mathbb{G}$ is uniformly connected. So (7.3),(7.4),(7.13) is an asymptotically autonomous system. The limiting autonomous system is obtained by defining $\bar{F}_k = Q_k^T\,\bar{Y}\,e^{t[\omega_0]^\wedge}$ whose dynamics

$$\begin{aligned} \tfrac{d}{dt}\bar{F}_k &= \bar{F}_k[\omega_0]^\wedge - [\omega_k]^\wedge\bar{F}_k \\ J_k\tfrac{d}{dt}\omega_k &= J_k\tfrac{d}{dt}\omega_k^{(d)} + \mu(\omega_k^{(d)} + \omega_0 - \omega_k) \\ \omega_k^{(d)} &= \tfrac{\gamma_S}{2}\,[\bar{F}_k - \bar{F}_k^T]^\vee \end{aligned}$$

are decoupled for the individual agents $k = 1, 2, ..., N$.

According to Proposition 2.3.6, the limit set of the original system is contained in the chain recurrent set of the limiting system. In analogy with case (a), define

$$W_k = -\mathrm{trace}(\bar{F}_k) + \tfrac{\kappa}{2}(\omega_k^{(r)} - \omega_k^{(d)})^T J_k(\omega_k^{(r)} - \omega_k^{(d)})\,.$$

Computing $\frac{d}{dt}\mathrm{trace}(\bar{F}_k) = \frac{2}{\gamma_S}(\omega_k^{(d)})^T(\omega_k - \omega_0)$ and after the same basic calculations as for (a),

$$\tfrac{d}{dt}W_k = -(\mu\kappa - \tfrac{1}{\gamma_S})\|\omega_k^{(r)} - \omega_k^{(d)}\|^2 - \tfrac{1}{\gamma_S}(\|\omega_k^{(r)}\|^2 + \|\omega_k^{(d)}\|^2) \quad \leq 0$$

with $\kappa > \frac{1}{\mu\alpha}$. Therefore LaSalle's invariance principle implies that the positive limit set $L^+(x)$ of any point $x := (\bar{F}_k, \omega_k)$ in the state space of the autonomous limit system only contains points where $\omega_k = \omega_0$ and $\omega_k^{(d)} = 0$. Denote the set of all these limit points by $\mathcal{E}$.

Proposition 2.3.7 is used to show that all chain recurrent points belong to $\mathcal{E}$. Assume that $x \notin \mathcal{E}$. Define $\nu_1 := \max_{y \in L^+(x)}(W_k(y)) < W_k(x)$ and $S := \{y : W_k(y) \leq \nu_1\}$. S contains $L^+(x)$ but not x. From Lemma A.1 in the Appendix, it follows that for any $\bar{Y}$, the function $\text{trace}(\bar{F}_k)$ can only take a finite number of values in $\mathcal{E}$, because $\omega_k^{(d)} = \frac{\gamma_S}{2}[\bar{F}_k - \bar{F}_k^T]^\vee = 0$ implies $\text{grad}_{Q_k}(\text{trace}(\bar{F}_k)) = 0$. Thus W_k as well can only take a finite number of values in $\mathcal{E}$. It follows that there exists $\nu_2 > \nu_1$ such that $W_k(y) \in (\nu_1, \nu_2) \Rightarrow y \notin \mathcal{E}$. Choose an ε_1-neighborhood S_{ε_1} of S and define $\rho = \min_{\{y=(\bar{F}_k,\omega_k)\} \setminus S_{\varepsilon_1}}(W_k(y)) > \nu_1$. Select an ε_2-neighborhood $S_{\varepsilon_2} = \{y : W_k(y) < \min(\{\nu_2, \rho\})\}$. Starting in S_{ε_2} ensures staying in S_{ε_1} so S is locally stable. Every point of S_{ε_2} must converge to a point of $\mathcal{E} \cap S_{\varepsilon_2} \subseteq \mathcal{E} \cap S$ so S is locally asymptotically stable. Thus S is an asymptotically stable set containing $L^+(x)$ but not x. Then from Proposition 2.3.7, x cannot be chain recurrent, so the chain recurrent set only contains points of $\mathcal{E}$.

This characterizes the limit set of the original system. The analysis in the Appendix, Lemma A.1, of critical points of a function $\text{trace}(Q^T B)$, $B \in \mathbb{R}^{3\times3}$ over $Q \in SO(3)$ shows that, for almost all initial conditions, $\bar{Y}$ is such that the critical points of $\text{trace}(\bar{F}_k)$ reduce to one unique maximum $\bar{F}^*$ and three unstable points. All solutions starting outside these three points and their stable manifolds converge to $\bar{F}_k = \bar{F}^*$, implying $Q_k = Q^* e^{[\omega_0]^\wedge t}\ \forall k \in \mathcal{V}$. □

An important remark for Proposition 7.2.1 (a) is that if $\omega_0 \neq 0$, then the set of configurations different from synchronization appearing in the limit set can be *smaller* than the set of critical points of V_L, which is the set of equilibria for the velocity-controlled algorithm (6.5). Indeed, the relative orientation $Q_k^T Q_j$ of rigid bodies k and j with identical angular velocity $\omega_0 \neq 0$ in body frame is *varying* if $Q_k\omega_0 \neq Q_j\omega_0$. Therefore, there can be sets of consensus configurations, stable under (6.5), which are not compatible with $\omega_k = \omega_0\ \forall k$. Such "incompatible consensus configurations" are not stabilized with (7.13).

Unstable situations for Proposition 7.2.1 (b) include some Q_k being turned by exactly π with respect to the projection of agreement matrix $\bar{Y}$ on $SO(3)$, and the rare cases where the Y_k converge to a matrix whose projection on $SO(3)$ is not unique.

Proposition 7.2.2: *Consider a swarm of N rigid bodies communicating along the edges of a δ-digraph $\mathbb{G}$ and whose orientation dynamics follow (7.3),(7.4) with control torque (7.14).*

(a) *Assume that $\mathbb{G}$ is fixed and undirected, $\omega_k^{(d)}$ is defined by (7.5)* where ω_0 is aligned with the principal axis *corresponding to J_{ki} of the rigid bodies, and*

$$\mu > (\sqrt{6}\alpha d_k^{(i)} J_{k1} + \tfrac{3J_{k1}+J_{ki}}{2}\|\omega_0\|) \tag{7.18}$$

for all agents k, where $d_k^{(i)}$ is the in-degree of k. Then the rigid bodies converge to a set where $\omega_k = \omega_0\ \forall k \in \mathcal{V}$ and the orientations are at a critical point of V_L defined in Section 5.3. The only stable solutions regarding relative orientations are consensus configurations of $\mathbb{G}$ that are compatible with $\omega_k = \omega_0\ \forall k \in \mathcal{V}$, *and if $\mathbb{G}$ is the complete graph or a tree, then the only stable solution is attitude synchronization.*

(b) *Assume that $\mathbb{G}$ is piecewise continuous and uniformly connected, $\omega_k^{(d)}$ is defined by (7.11),(7.12) emphwhere ω_0 is aligned with the principal axis corresponding to J_{ki} of*

the rigid bodies, and

$$\mu > (\sqrt{2}\gamma_S J_{k1}\|F\| + \tfrac{3J_{k1}+J_{ki}}{2}\|\omega_0\|) \tag{7.19}$$

for all agents k and for all t, where $\|F\|$ is the Frobenius norm of matrix $F \in \mathbb{R}^{3\times 3}$. Then for almost all initial conditions $F_k(0)$, $Q_k(0)$, $\omega_k(0)$ for which $\|F_k(t)\| < \|F\|$ $\forall t$ and $\forall k \in \mathcal{V}$, the solutions converge to attitude synchronization with $\omega_k = \omega_0$ $\forall k \in \mathcal{V}$.

Proof: For (a), proceeding as for the proof of Proposition 7.2.1 yields

$$\tfrac{d}{dt}W = \tfrac{d}{dt}W_{(7.17)} + \sum_{k=1}^{N} \kappa(\omega_k^{(r)} - \omega_k^{(d)})^T((J_k\omega_k) \wedge \omega_k) \tag{7.20}$$

where $\frac{d}{dt}W_{(7.17)}$ denotes the evolution of W with control torque (7.13), given by (7.17). The last term, of indefinite sign, is overcome by the definite terms of $\frac{d}{dt}W_{(7.17)}$ if the condition on μ holds, see Appendix A.3. The proof then concludes as for Proposition 7.2.1 (a).

The proof for (b) adapts the proof of Proposition 7.2.1 in a similar way. The bound on μ is derived as in Appendix A.3 for case (a), observing that now $\|\omega_k^{(d)}\| \leq \sqrt{2}\gamma_S\|F_k\|$. □

Proposition 7.2.2 allows to impose a rotation ω_0 around a principal axis only. Thus the imposed final motion is always an equilibrium of the free rigid body dynamics. This is probably the most useful case in practice since maintaining other motions would require persistent control torques.

The bound in Proposition 7.2.2 (b) involves auxiliary variable $F_k(t)$. The values of $\|F_k\|$ are ensured to stay bounded because, for whatever $\mathbb{G}$, at any time the values of the Q_kZ_k lie in the convex hull of the initial values $\{Q_1Z_1(0), Q_2Z_2(0), ...Q_NZ_N(0)\}$, and $\|F_k\| = \|Z_k\| = \|Q_kZ_k\|$.

Overall, the consensus tracking approach has the advantage of being flexible because it can be adapted, as well to various algorithms defining the desired velocity (i.e. alternative consensus algorithms), as to various mechanical models by using tracking controllers from the literature. However, since it essentially counters the natural dynamics of the system, it is expected to be less power-efficient and less robust than other control methods, like the following energy shaping approach. Another point is that it imposes the final motion of the swarm. This is also further examined in the following section.

7.3 Energy shaping

The idea of *energy shaping* for control design of mechanical systems is to "shape" the potential and kinetic energy of the system in order to make the desired behavior a stable solution of the "shaped" system. Control forces and torques implement the difference between the "shaped" energy and the original energy. An additional control term is designed to *dissipate* the shaped energy of the system, such that the desired behavior becomes *asymptotically* stable. An advantage observed in practice for energy shaping with respect to other control methods is a better robustness.

Early work using the energy shaping method builds artificial potentials for robotic navigation and obstacle avoidance [67, 114]. Spacecraft control uses potential [85] and kinetic [11] energy shaping. Potential shaping is used in [74] for stabilization of rigid bodies in $SE(3)$.

Energy shaping is used for synchronization of mechanical system networks in [96, 99, 100] and applied to swarms on $SO(3)$ and $SE(3)$ in [48, 96, 97, 98, 135]. Kinetic energy shaping can transform any principal axis into the short axis [98, 135]; this part is ignored here. The following first shows how potential energy shaping in [96, 97, 98, 135], and less formally with local results in [109], achieves attitude synchronization in a framework linked to the developments of the present dissertation. It then proposes two extensions of existing results.

1. In [98, 135], the dissipative control term, based on angular velocities, imposes the final motion of the swarm. By only using *relative* angular velocities $Q_j\omega_j - Q_k\omega_k$ between agents, any synchronized free rigid body motion is a solution for the controlled swarm.
2. The results in [98, 135] are valid for fixed, undirected communication graph $\mathbb{G}$. The consensus strategy with auxiliary variables from Section 6.2.1 allows to design an energy shaping algorithm for attitude synchronization with directed and time-varying $\mathbb{G}$.

The convergence property for the first extension is not as strong as for the other controllers of the present chapter. An example shows that there are indeed limitations for controlling system (7.3),(7.4), which is not invariant with respect to absolute velocity ω_k, with feedback torques that only use relative velocities. This illustrates the relevance of the formal *reduction techniques* mentioned in Section 7.1 (see [48, 135] for a specific treatment of $SO(3)$).

For simplicity, assume that $\mathbb{G}$ is an unweighted graph. Write $u_k = u_k^{(P)} + u_k^{(D)}$, where $u_k^{(P)}$ denotes control torques implementing the artificial potential and $u_k^{(D)}$ denotes control torques implementing dissipation. The artificial energy is $H = T + V$, where kinetic energy $T = \sum_k T_k = \sum_k \frac{1}{2}\,\omega_k^T J_k \omega_k$ is the energy of free rigid bodies and V is the artificial potential to be designed. According to classical mechanics, $[u_k^{(P)}]^\wedge = -\mathrm{grad}_{Q_k}(V)$ and $\frac{d}{dt}H = \sum_k (u_k^{(D)})^T\,\omega_k$. The total angular momentum of the swarm is $M = \sum_k Q_k J_k \omega_k$ and $\frac{d}{dt}M = \sum_k Q_k u_k$.

7.3.1 An existing attitude synchronization algorithm

The artificial potential used in [96, 97, 98, 135] turns out to be equivalent to the specific form on $SO(3)$ of the cost function proposed for consensus algorithm desing in Chapters 5 and 6,

$$V = \tfrac{\sigma}{2} \sum_k \sum_{j \rightsquigarrow k} \mathrm{trace}(Q_k^T Q_j) \quad , \quad \sigma < 0\,. \tag{7.21}$$

The change of sign with respect to (5.13) makes attitude synchronization a *global minimum* of V; the latter can have local minima when G is not a tree or complete graph, as discussed in Chapter 5. In fact, a specific version of the same potential can also be found in [22], where one actual agent interconnected to one virtual (reference) agent is considered for the design of attitude tracking algorithms. The author of [109] uses a similar potential, but locally, with the quaternion representation. The gradient of V is computed in Section 6.1, yielding for $\mathbb{G}$ undirected

$$u_k^{(P)} = -[\mathrm{grad}_{Q_k}(V)]^\vee = -\sigma \sum_{j \rightsquigarrow k} [Q_k^T Q_j - Q_j^T Q_k]^\vee \quad , \quad k = 1, 2, ..., N\,. \tag{7.22}$$

When $u_k^{(D)} = 0$, attitude synchronization with rotation around the short axis is (Lyapunov) stable. *Asymptotic* stability requires $u_k^{(D)}$ to decrease H. In [96, 98], exponential stabilization is only considered in a context where V contains an additional term *aligning the*

short axis with a specific direction in inertial space. This does not satisfy invariance of the swarm's behavior with respect to a uniform rotation $(Q_1, Q_2, ..., Q_N) \to (QQ_1, QQ_2, ..., QQ_N)$. A dissipative torque suggested in [96, 135] and used in [109] is simply a drag on the (absolute) angular velocity:

$$u_k^{(D)} = -\gamma\, \omega_k \quad , \quad \gamma > 0\,, \;\; k = 1, 2, ..., N\,. \tag{7.23}$$

Torque u_k defined as the sum of (7.22) and (7.23) drives the agents to consensus configurations with *zero velocity* $\omega_k = 0$. It may be possible to design a variant achieving a specified nonzero velocity ω_0, as in Section 7.2.

Proposition 7.3.1: *(combining [96, 109, 135]) Consider a swarm of N rigid bodies communicating along the edges of a fixed undirected graph $\mathbb{G}$ and whose orientation dynamics follow (7.3),(7.4) with control torque $u_k = u_k^{(P)} + u_k^{(D)}$ whose terms are given by (7.22) and (7.23). Then the rigid bodies converge to a set where $\omega_k = 0\ \forall k \in \mathcal{V}$ and the orientations are at a critical point of V defined in (7.21). The only stable solutions regarding relative orientations are consensus configurations of $\mathbb{G}$, and if $\mathbb{G}$ is the complete graph or a tree, then the only stable solution is attitude synchronization.*

Proof: Artificial energy H can be used as a Lyapunov function, leading to

$$\tfrac{d}{dt} H = -\gamma \ \textstyle\sum_k \|\omega_k\|^2 \leq 0$$

and according to LaSalle's invariance principle, the limit set of the system is contained in the largest invariant set where $\omega_k = 0\ \forall k \in \mathcal{V}$. But if $\omega_k = 0$ and $u_k^{(P)} \neq 0$, then the right side of (7.4) is nonzero and ω_k will be driven away from zero. Therefore, an invariant set where $\omega_k = 0\ \forall k \in \mathcal{V}$ can only contain points where $u_k^{(P)} = 0\ \forall k$ as well. This corresponds to critical points of V. Stable configurations correspond to minima of V, which taking into account the developments of Chapter 5 concludes the proof. □

7.3.2 Extension 1: relative angular velocities

Dissipation (7.23), although it preserves symmetry with respect to *orientation* of the agents, imposes their asymptotic *motion*. As an alternative, it is possible to imagine a dissipative term which drives agent velocities *towards each other* instead of towards zero, comparing $Q_k\omega_k$ to the $Q_j\omega_j$ (inertial frame), or equivalently ω_k to the $Q_k^T Q_j \omega_j$ (body frame). The corresponding torque is

$$u_k^{(D)} = \gamma \sum_{j \rightsquigarrow k} (Q_k^T Q_j \omega_j - \omega_k) \quad , \quad \gamma > 0\,, \;\; k = 1, 2, ..., N\,. \tag{7.24}$$

The advantage of such "relative dissipation" over "absolute dissipation" as in (7.23) is that, assuming $J_k = J\ \forall k \in \mathcal{V}$, *any free rigid body motion of synchronized agents is a solution of the closed-loop system.* General forms and convergence results for relative dissipation are proposed in [58] based on reduction techniques; they require consecutive Poisson brackets of allowed torques to restore full rank, which is not true here.

Attitude synchronization is more difficult with (7.24) than with (7.23) because the u_k only influence *relative* velocities, while the rigid body dynamics still depend on ω_k. The following result illustrates typical difficulties caused by the remaining angular velocity in the dynamics.

Proposition 7.3.2: *Consider a swarm of N rigid bodies communicating along the edges of a fixed, connected and undirected graph $\mathbb{G}$ and whose orientation dynamics follow (7.3),(7.4) with control torque $u_k = u_k^{(P)} + u_k^{(D)}$ whose terms are given by (7.24) and $u_k^{(P)} = -[\mathrm{grad}_{Q_k}(V)]^\vee$, where V is a bounded potential.*

(a) *Regardless of V, velocities in inertial frame $Q_k\omega_k$ globally asymptotically synchronize — but in general not to a* fixed *value.*

(b) *For identical rigid bodies ($J_k = J\ \forall k \in \mathcal{V}$) and V defined by (7.21), angular momentum M is conserved. Given $M_{max} \in \mathbb{R}_{>0}$, there exists $\sigma^* < 0$ (depending on N, J, $\mathbb{G}$ and M_{max}) such that for $|\sigma| > |\sigma^*|$, the set of free rigid body motions with synchronized attitudes $Q_k(t) = Q_j(t)\ \forall j,k$ and with $\|M\| \in (0, M_{max})$ is locally asymptotically stable.*

Proof: Artificial energy H is used as a Lyapunov function. This yields

$$\begin{aligned} \tfrac{d}{dt}H &= \gamma \sum_k \sum_{j\rightsquigarrow k} (\omega_k)^T (Q_k^T Q_j \omega_j - \omega_k) \\ &= \gamma \sum_k \sum_{j\rightsquigarrow k} (Q_k\omega_k)^T (Q_j\omega_j - Q_k\omega_k) = -\gamma\, (\Omega^a)^T (L \otimes I_3)\, \Omega^a \end{aligned}$$

where Ω^a is the $3N$-vector containing all $Q_k\omega_k$ stacked, and L is the Laplacian of $\mathbb{G}$. For $\mathbb{G}$ undirected and connected, L is positive semidefinite and its kernel reduces to $c\,\mathbf{1}_N$ with $c \in \mathbb{R}$. Thus H decreases unless all $Q_k\omega_k$ are equal, so LaSalle's invariance principle ensures that the swarm converges to an invariant set, under dynamics (7.3),(7.4) with $u_k = u_k^{(P)}$, where $Q_k\omega_k = Q_j\omega_j\ \ \forall j,k \in \mathcal{V}$. This proves part (a).

For part (b), conservation of M is equivalent to $\sum_k Q_k u_k = 0$ which is easy to verify. For synchronization of the Q_k, the proof is in two steps. Define the set S_{M^*} of all free rigid body motions with synchronized attitudes $Q_k(t) = Q_j(t)\ \forall j,k$ and total angular momentum $\|M^*\| < M_{max}$. First, show that given a neighborhood $W \ni (Q_1, Q_2, ..., Q_N, \omega_1, \omega_2, ..., \omega_N)$ of S_{M^*}, there exist $|\sigma_1|$ and a neighborhood U of S_{M^*} such that starting in U implies staying in W if $|\sigma| > |\sigma_1|$. Then consider solutions of (7.3),(7.4) with identical $Q_k\omega_k$ and $u_k = u_k^{(P)}$ from (7.22). Show that there exist $|\sigma_2|$ and a neighborhood W_1 of S_{M^*} such that for $|\sigma| > |\sigma_2|$, solutions that stay in W_1 are necessarily in S_{M_0} for some $M_0 \in (0, M_{max})$. Taking $W = W_1$ and $|\sigma^*| > \max(|\sigma_1|, |\sigma_2|)$ concludes the proof; see Appendix A.4 for details. □

Several comments are in order about Proposition 7.3.2.

- Proposition 7.3.2 (a) still holds for time-varying (uniformly connected) and *balanced* directed graphs, because $x^T L\, x$ is still non-negative $\forall x$ in this case.
- When the swarm is synchronized, all control torques u_k vanish. Hence, the (unimposed) motion of the synchronized swarm can indeed be any free rigid body motion.
- Proposition 7.3.2 (b) is a local result. However, simulations indicate a large basin of attraction U, see for instance Figure 7.1. The proof in Appendix A.4 contains three conditions for U. Conditions with κ and β basically impose $\|M\|$ lower and upper bounded and $\|\omega_k\|$ upper bounded; for "infinite σ", this still allows almost any initial condition. The critical constraint is with ε chosen to bound high order terms in (11.37). For "infinite σ" relevant high-order terms are on the left; they are like $(\sin(\theta))^2 - \theta^2$ around $\theta = 0$. Then it is sufficient that all Q_k are inside a closed geodesic ball of radius $< \pi/2$, which is just smaller than the maximal convex set of $SO(3)$. This is consistent with synchronization being the only minimum of V with all Q_k within a convex set.
- The bound on $|\sigma|$ reflects that the controller must overcome unknown "perturbations" $(J\omega_k) \wedge \omega_k$. Sliding mode control, like changing $u_k^{(P)}$ to $\mathrm{grad_k}(V)/\|\mathrm{grad_k}(V)\|$, is similar to "infinite σ" near synchronization. However, the resulting chattering depends

on controller parameters, and ω_k-dependent bounds will still be required to stay in a neighborhood of synchronization where the "perturbations" do not desynchronize the system. This even gets more difficult to analyze since $\frac{d}{dt}M \neq 0$ and $\frac{d}{dt}H \not\leq 0$.

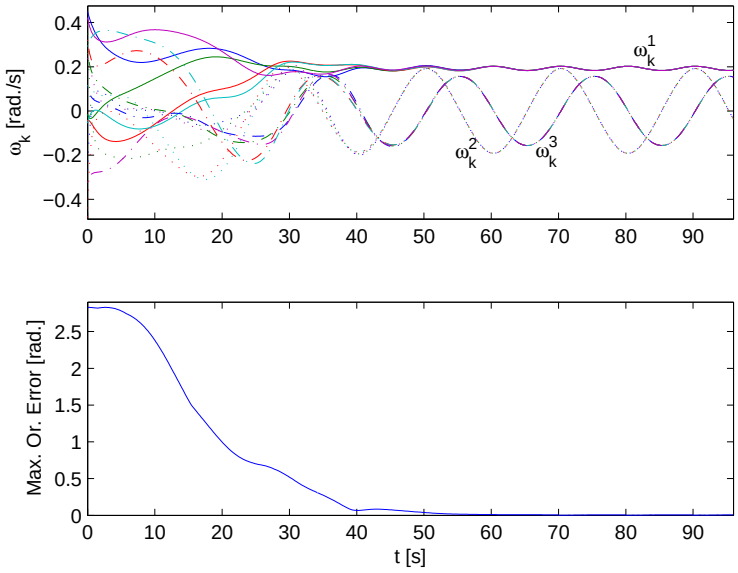


Figure 7.1: Simulation of the energy shaping control algorithm with conservative torque (7.22) and relative dissipation torque (7.24) for a swarm of five rigid bodies. The interconnection graph $\mathbb{G}$ is an undirected ring. Initial orientations and velocities are randomly chosen. Angular velocities $\omega_k = (\omega_k^1\ \omega_k^2\ \omega_k^3)$ (top) are represented, as well as the orientation error (bottom) defined as the maximal geodesic distance (Euler rotation angle) between any pair of agents.

With no condition on σ, there are situations arbitrarily close to synchronization but from where synchronization is never reached, as illustrated in the following example.

Ex. 7.3.3: a situation arbitrarily close to, but never reaching synchronization for too small σ: Take two agents A and B synchronized and rotating around e_3 with velocity Ω, where e_1, e_2, e_3 denote principal axes of $J_1 > J_2 > J_3$. Now consider the situation where (see Figure 7.2), with respect to this synchronized state, A and B are tilted by ϕ and $-\phi$ respectively around e_2, with ϕ arbitrarily small; they still rotate with angular velocity $Q_A\omega_A = Q_B\omega_B = \Omega$ around the axis aligned with the initial synchronized e_3, which now makes an angle ϕ with actual axes e_{3A} and e_{3B}. Then $(J\omega_k) \wedge \omega_k$ pulls A and B further apart, while $u_k^{(P)}$ pulls them together. For a particular ratio of $\|\Omega\|^2$ and $|\sigma|$, both effects exactly cancel, such that angular velocity $Q_A\omega_A = Q_B\omega_B = \Omega$ remains unchanged, the tilt is maintained and the agents never synchronize. For lower $|\sigma|$, the repulsive term is dominant and the agents are even (initially) driven apart.

◇

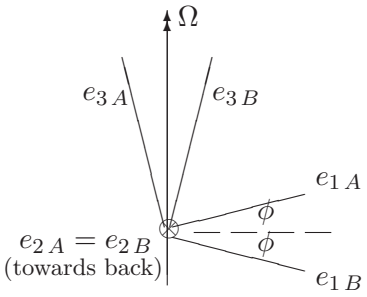


Figure 7.2: Two rigid bodies in a situation from which (7.22),(7.24) do not synchronize attitudes for small σ. All vectors lie in the same plane, except $e_{2\,A} = e_{2\,B}$ which is perpendicular to the page.

7.3.3 Extension 2: directed and varying graphs

The algorithms in Sections 7.3.1 and 7.3.2 require a fixed, undirected graph $\mathbb{G}$. It is also possible to adapt the synchronization strategy with auxiliary variables to the energy shaping framework. As in the settings of Sections 6.2.1 and 7.2, auxiliary variables $Z_k \in \mathbb{R}^{3\times 3}$ apply the consensus algorithm

$$\tfrac{d}{dt} Z_k = \beta \sum_{j \rightsquigarrow k} ((Q_k^T Q_j) Z_j - Z_k) \quad - [\omega_k]^\wedge Z_k \quad , \quad k = 1, 2, ..., N \, , \tag{7.25}$$

such that the $Y_k := Q_k Z_k$ asymptotically agree on a common matrix, $Y_k = \bar{Y} \ \forall k \in \mathcal{V}$. Then an artificial potential V is built as the sum of individual agent potentials V_k. The form of V_k is inspired from Chapter 5. The Frobenius distance in $\mathbb{R}^{3\times 3}$ from Y_k to $Q \in SO(3)$ is

$$\|Y_k - Q\|^2 = \mathrm{trace}((Y_k - Q)^T (Y_k - Q)) = 3 + \mathrm{trace}(Y_k^T Y_k) - 2\,\mathrm{trace}(Q_k^T Y_k) \, .$$

Therefore $V_k(Q_k) = \sigma\,\mathrm{trace}(Q_k^T Y_k)$ with $\sigma < 0$ is chosen. At the minimum of V_k, orientation Q_k is the projection of Y_k onto $SO(3)$, as characterized in Proposition 5.1.5; this projection is unique for almost all Y_k. Then when $Y_k = \bar{Y} \ \forall k \in \mathcal{V}$ (and since $\bar{Y}$ takes the same generic values as the Y_k), all Q_k would be equal. The associated torque is

$$u_k^{(P)} = -\sigma\,[Z_k - Z_k^T]^\vee \quad , \quad \sigma < 0 \, , \quad k = 1, 2, ..., N \, . \tag{7.26}$$

The fact that Q_k tracks $Y_k = Q_k Z_k$ defined by (7.25) implies that synchronization takes place at a fixed orientation defined by $\bar{Y}$, with no possible motion of the synchronized agents, whatever dissipative torque $u_k^{(D)}$ is used. This is reflected by the presence of ω_k in (7.25). In this context, there is no reason to head for the difficulties encountered in Section 7.3.2 with relative dissipation (7.24), so $u_k^{(D)}$ is defined by (7.23).

Proposition 7.3.4: *Consider a swarm of N rigid bodies communicating along the edges of a piecewise continuous and uniformly connected graph $\mathbb{G}$ and whose orientation dynamics follow (7.3),(7.4) with control torque $u_k = u_k^{(P)} + u_k^{(D)}$ whose terms are given by (7.25),(7.26) and (7.23). Then attitude synchronization with $\omega_k = 0 \ \forall k$ is almost globally asymptotically stable.*

Proof: The proof idea is similar to Proposition 7.2.1 (b).

The $Y_k := Q_k Z_k$ evolve independently of the agents' motions and exponentially convergence to a common constant $\bar{Y}$ for $\mathbb{G}$ uniformly connected. Then (7.3),(7.4),(7.23),(7.26) form an asymptotically autonomous system where agents are decoupled; the limiting (autonomous) system is obtained by replacing Z_k with $Q_k^T \bar{Y}$. According to Proposition 2.3.6, the limit set of the original system is contained in the chain recurrent set of the limiting system. The limiting system for individual agent k is of the "shaped energy" form with H bounded below and $\frac{d}{dt}H = -\gamma \|\omega_k\|^2$. A LaSalle argument on H as for Proposition 7.3.1 shows that the positive limit set $L^+(x)$ for the autonomous system of any point $x := (Q_k, \omega_k)$ only contains equilibria, where $\omega_k = 0$ and Q_k is at a critical point of V_k; denote the set of all these equilibria by $\mathcal{E}$. Proposition 2.3.7 can be used exactly as in the proof of Proposition 7.2.1 (b) to show that a point which does not belong to $\mathcal{E}$ cannot be chain recurrent. Thus the limit set of the original system reduces to the equilibria of the limiting system. The analysis of critical points of function trace$(Q_k^T B)$, $B \in \mathbb{R}^{3\times 3}$ over $Q_k \in SO(3)$ made in Lemma A.1 of the Appendix shows that, for almost all initial conditions, $\bar{Y}$ will be such that the set of critical points of $V_k(Q_k)$ only contains one unique minimum Q^* and three unstable points. All solutions starting outside these three points and their stable manifolds converge to $Q_k = Q^* \ \forall k \in \mathcal{V}$, implying attitude synchronization. □

As for Proposition 7.2.1 (b), unstable situations that do not converge to synchronization consist of the rare cases where the Y_k converge to a matrix whose projection on $SO(3)$ is not unique, and configurations where some Q_k are turned by exactly π with respect to the projection of the agreement matrix $\bar{Y}$ on $SO(3)$.

Different adaptations of (7.25) have been explored to make use of *relative* angular velocities only. The hope is that combining the extensions of Sections 7.3.2 and 7.3.3 could lead to an algorithm which works with directed and time-varying $\mathbb{G}$ and leaves the possibility for the synchronized swarm to move according to any free rigid body motion. Unfortunately, conclusive convergence results were obtained neither analytically, nor in simulations.

O.Q.: It is currently unknown whether and how it is possible to combine the extensions of Section 7.3.2 — using only *relative* angular velocities $Q_j\omega_j - Q_k\omega_k$ — and of Section 7.3.3 — using a consensus algorithm on auxiliary variables — in order to obtain a control algorithm that would (i) achieve almost global attitude synchronization for varying and directed interconnection graphs and (ii) at the same time, allow the final motion of the synchronized swarm to be any free rigid body motion.

Recapitulation

Part II of the dissertation is devoted to the study of position coordination on connected compact homogeneous manifolds. This involves defining properties for specific sets of relative positions ("configurations") on such manifolds, and designing and analyzing algorithms to reach such configurations. Part II extends Part I in several directions.

It starts by defining the induced arithmetic mean of N points on an embedded connected compact homogeneous manifold $\mathcal{M}$. Although the induced arithmetic mean differs from the traditional Karcher mean, it has a clear geometric meaning with the advantage of being easily computable — see analytical solutions for $SO(n)$ and $Grass(p,n)$. Then, specific configurations for points on $\mathcal{M}$ are defined in direct relation with the induced arithmetic mean. Specifically, consensus is a situation where each agent is "as close as possible to" — i.e. at the induced arithmetic mean of — its neighbors, and anti-consensus is a situation where each agent is "as far as possible from" — i.e. at the "anti"-induced arithmetic mean of — its neighbors. In addition, the notion of balancing, first mentioned in [131] for the circle, is extended to connected compact homogeneous manifolds. Consensus for the equally-weighted complete graph is equivalent to synchronization. Likewise, it appears in simulations that anti-consensus for the equally-weighted complete graph leads to balancing (if N is large enough), even though this could not be proved. A cost function is built to formulate the achievement of consensus and related configurations as a distributed optimization problem.

In a next step, gradient algorithms are derived for fixed undirected interconnection graphs; (anti-)consensus configurations are their only stable equilibria. The global convergence properties of similar algorithms when the graph is allowed to be directed and/or to vary are mostly open. Therefore new algorithms are introduced to theoretically ensure that synchronization or balancing is attained with the weaker requirement of uniformly connected interconnection graphs. These algorithms, inspired by the dynamic controller of Section 4.3 in Part I, use auxiliary variables to overcome the problems related to the nonconvex manifold geometry; an efficient way of communicating auxiliary variables between agents remains an open issue when $\mathcal{M}$ is not $SO(n)$. Other algorithms for global synchronization are briefly investigated as well.

Running examples $SO(n)$ and $Grass(p,n)$ illustrate the validity of the discussion so far and provide geometric insight. In addition, it is shown how the models and results obtained by applying the framework to the circle are strictly equivalent to existing models and results of Part I, thereby drawing a link from the present discussion to the vast literature about synchronization and "spreading" problems on the circle.

The closing chapter illustrates how to use these tools with more complex dynamics. To this end, it revisits the popular subject of rigid body attitude synchronization, considering the mechanical dynamics of rigid body rotation (Euler equations), in a setting that is invariant with respect to absolute orientation; this mainly means that the agents have no

common reference to track, in contrast to a large part of the literature. Specifically, two control methods are examined to obtain attitude synchronization: consensus tracking and energy shaping. Consensus tracking is a backstepping approach, in which the velocities of the consensus algorithms of the previous chapter are assigned as desired (reference) velocities to an appropriate mechanical tracking controller. Energy shaping uses the cost function, which formulates consensus as an optimization problem, as an artificial potential, and with appropriate dissipation drives the mechanical system to a minimum of this potential. The latter approach has already been investigated by [96, 97, 98]. The present dissertation extends this work in two separate ways. It first examines what convergence can be achieved by only using relative angular velocities in the control, which allows the final motion to be any free rigid body motion instead of imposing it in advance. Then it shows that the consensus strategy with auxiliary variables can be adapted to energy shaping in order to obtain almost-global attitude synchronization for uniformly connected interconnection graphs.

As an overall conclusion of Part II, it can be said that the "global consensus problem" has been appropriately generalized from vector spaces to manifolds: definitions and properties for specific configurations are proposed, and corresponding algorithms to reach them are provided, either retaining some manifold-related convergence specificities or using particular tricks to obtain as strong convergence properties as on vector spaces.

The original contribution in Part II includes basically all the material of Chapters 5 and 6. Although the concepts introduced in Chapter 5 are not far-fetched, it seems that they have never been formalized before; besides the circle, which is amply discussed in Part I, there are just a few authors treating $SO(3)$ in a way that can be linked to the present general theory. As a result, the associated algorithms in Chapter 6 are original as well, in particular the generalization of the global synchronization algorithms of Chapter 4. The topic of Chapter 7 has attracted more attention, but mostly in conjunction with reference tracking. The content of Section 7.2 is original, although the proposed approach is rather classical in nonlinear control. The energy shaping approach of Section 7.3 is more popular, and results close to the proposed ones have been published, as duly documented in the literature review. Here, beyond the two separate extensions, the contribution is also to show how existing control algorithms fit into the general framework in the rest of Part II.

The original content of Part II is published (Chapters 5 and 6) in [118, 117] and (Chapter 7) in [119, 120, 121].

Part III

Coordinated motion on Lie groups

Part III is inspired by the work published in [130, 131, 132] and [123, 126] about coordinated rigid body motion in two and three dimensions respectively. Discussions with the authors of these papers — Dr. Luca Scardovi, Prof. Derek Paley and Prof. Naomi Leonard — were certainly beneficial for the elaboration of the following general theory. The latter has been developed in collaboration with Dr. Silvère Bonnabel.

Introduction The objective in the third part of this dissertation is to provide a unified geometric framework for *coordinated motion* in a swarm of interacting agents on Lie groups. In the terms of Section 1.2 in the general Introduction, Part III focuses on "velocity coordination", in contrast to Parts I and II which are concerned with "position coordination", i.e. achieving specific *configurations*. Example 1.2.7 illustrates the relevance of the coordinated motion problem on the sphere. In the remainder of Part III, *coordinated motion* is sometimes just called *coordination* to shorten denominations.

The present chapters consider a general theory to define coordinated motion(s) and analyze ensuing conditions and properties. Then they introduce a precise method to design control laws achieving coordinated motion in a set of autonomous, interacting agents with fully actuated or underactuated simple integrator dynamics. This formalizes and extends existing results for collective motion of rigid bodies in the plane (as in [63, 130, 131, 132]) and in three-dimensional space (as in [64, 123, 126]) to a common framework that can be applied to a broad variety of practical settings. The goal is thus to provide a systematic approach that might be helpful to study coordinated motion in several situations.

The configuration space for the agents is assumed to be a Lie group. Lie groups possess strong symmetry properties, allowing most importantly (i) to combine two positions in order to build a third one, and (ii) to compare velocities at different locations in a common tangent plane. The first property allows a natural definition of "relative positions" on Lie groups; this directly leads to definitions of coordinated motions as motions where relative positions remain fixed. The second property allows to advantageously formulate the conditions for coordination in terms of velocities, which are used for controller design. As a result, coordination can be studied in a systematic way once the Lie group geometry of the configuration space is known.

The Lie group setting captures practical situations for which "relative positions" and "motion in formation" are intuitively clear. Consider for instance the example of collective motion of vehicles in a vector space $\mathbb{R}^n$ (be it on the ground, under water, in the air, or in space). The vector space $\mathbb{R}^n$ itself is a trivial Abelian Lie group, but the orientation of the vehicles in $\mathbb{R}^n$ involves the Lie group $SO(n)$ of rigid body rotations. Orientations are not only important to model the full configuration of a real rigid body, but they must also be introduced in order to properly characterize the evolution of the trajectory in a Serret-Frenet frame. The intuitive definitions of "relative positions" and "motion in formation" in this context are captured by the geometry of the Lie group $SE(n)$ of rigid body translations and rotations coupled in a particular way. The nonlinear Lie groups $SO(n)$ and $SE(n)$ are the most prominent examples of application of the present work. Importantly, unlike for Parts I and II, the Lie groups are not restricted to be compact.

Homogeneous manifolds different from Lie groups do not possess the rich structure of Lie groups, such that a similar systematic study cannot be carried out. However, it must be emphasized that often the intuitive meaning given to "motion in formation" on a homogeneous manifold is, as for vector spaces, captured by attributing not only a position on the manifold but also an "orientation" to the agents; in this way a Lie group is retrieved and the theory

applies. For instance, Example 8.2.10 in the following provides a rigorous Lie group framework to the intuitive discussion of coordinated motion on the sphere S^2, see Example 1.2.7.

Outline and Main points Global geometry and symmetry are fundamental in order to obtain a unified framework for coordinated motion. The symmetry of the Lie group setting enters at two points for coordinated motion. The first point is to formulate the objective; here invariance serves as a basis for the definition of what should be conserved as the swarm moves along on the Lie group manifold. The second point is in the control setting; here the Lie group structure is advantageous for proper algorithm design.

Part III provides a *unified geometric framework* for the study and design of coordinated motion on Lie groups. This is expected to facilitate the design of control laws for collective motion in several settings. The general theory comprises four points: (i) coordinated motions which have been previously considered for particular examples are formally defined from first principles and related consequences are examined; (ii) the particular notion of total coordination is proposed and several interesting related properties are highlighted; (iii) a systematic method is proposed to design control laws for interacting autonomous agents such that they asymptotically achieve coordinated motion in various settings (among others, taking underactuation into account); (iv) a geometric line of reasoning is proposed for the design and interpretation of control laws to reach coordinated motion in combination with specific configurations of the relative positions, as defined in Parts I and II.

Chapter 8 defines coordinated motion from first principles of symmetry, derives associated conditions on velocities and examines related implications. The Lie group structure yields two natural symmetry properties with respect to a uniform translation of all the agents: invariance with respect to right multiplication $(g_1, g_2, ..., g_N) \to (g_1 g, g_2 g, ..., g_N g)$ and invariance with respect to left multiplication $(g_1, g_2, ..., g_N) \to (g g_1, g g_2, ... g g_N)$, for any group element g. Section 8.1 shows how those two different invariance types lead to two different abstract definitions of relative positions — right-invariant and left-invariant — and two associated types of *coordinated motion*, defined as motions during which the relative positions remain constant. In practice, on physical interpretations of the Lie group, both types of coordination have a different intuitive meaning. *Total coordination* is defined as simultaneous right- and left-invariant coordination. Section 8.2 expresses the conditions for coordination in the associated Lie algebra, and thereby draws a direct link between coordinated motion on Lie groups and synchronization of velocities in a vector space. Left and right translations lead to two possible, but coupled viewpoints. It is then investigated how total coordination restricts compatible relative positions through a geometrically meaningful condition, involving the relation that couples the left- and right-invariant viewpoints on velocities. These properties are independent of the agents' dynamics.

Chapter 9 provides a systematic method to design control laws achieving coordination. As always in the present dissertation, communication among agents is restricted to a reduced set of links (characterized by the edges of a possibly directed and time-varying interconnection graph), and the agents have to reach relevant agreements in an invariant way, without any hierarchy or external reference. Simplified first-order dynamics are assumed for individual agents, but underactuation is explicitly modeled. Like for the achievement of particular configurations, considering simplified dynamics can be useful either to build a high-level "planning" controller or as a preliminary step towards an integrated mechanical controller. The control setting uses the convention that left-invariant variables represent quantities in

agent frame (in contrast to an inertial reference frame). In Section 9.1, control laws based on standard vector space consensus algorithms are given that achieve the easier tasks of right-invariant coordination and fully actuated left-invariant coordination on general Lie groups, for any initial condition. Section 9.2 then proposes a general method to design control laws that achieve total coordination of fully actuated agents when the communication links among agents are undirected and fixed; an extension to more general communication settings can be made along the lines of [132]. Total coordination of fully actuated agents is a rather academic problem, but in Section 9.3 the proposed design methodology is shown to apply to the practically most relevant problem of left-invariant coordination of underactuated agents; indeed, the latter task also requires to reach compatible relative positions. The proposed controller architecture consists of two steps, obtained by adding to the consensus algorithm a position controller derived from geometric Lyapunov functions. It is shown that the position controller is directly linked to the double bracket flows for gradient systems on adjoint orbits proposed in [19]. As a finale of the dissertation, Section 9.4 discusses how to combine the control method for "coordinating the motion" with the tools of Part II, such that the particular formation in which the agents move would be a consensus configuration.

At the end of Chapter 8, the abstract geometric concepts are illustrated on examples $SO(3)$, $SE(2)$ and $SE(3)$. The power of the geometric controller design method is illustrated with the same examples at convenient places in Chapter 9. This shows how controllers that have been previously proposed in the literature, on the basis of intuitive arguments for the particular applications, can be retrieved essentially systematically with the proposed general methodology.

Related literature As explained in the general Introduction (Chapter 1), coordinated motion on a vector space — only considering the translation symmetry, with no meaning given to rotations of agents or trajectories — is strictly equivalent to position synchronization on the same vector space from a geometric point of view: because a vector space can be identified with its tangent plane, coordinated motion simply amounts to consensus on the linear velocity. Thus the well-established results for synchronization on vector spaces, like [13, 94, 95, 102, 104, 143], cover this application. However, as mentioned in a previous paragraph, intuitive applications with "formations" moving on $\mathbb{R}^n$ are mostly implicitly considering motion on the nonlinear Lie group $SE(n)$, by allowing rotations as a meaningful symmetry.

The geometric viewpoint for dynamical systems on Lie groups is a well-studied subject; see basic results in [5, 60] for simplified dynamics like those considered in the present part of the dissertation, and [5, 22, 75, 83] for a geometric theory of *mechanical* systems on Lie groups. In particular, [60] discusses controllability for underactuated systems on Lie groups, among which the groups $SO(n)$ and $SE(n)$ considered for illustration in the present work. A general introduction to Lie groups is presented in Section 2.4 of the present dissertation. Close to the present paper in its geometric flavor, [16] builds invariant *observers* for systems with Lie group symmetries; observer design can be seen as two-agent leader-follower synchronization.

Recently, coordination has been investigated on $SE(2)$ (see [63, 131, 132]) and $SE(3)$ (see [48, 64, 126, 135]) in the underactuated setting of *steering control* where the linear velocity is fixed in the agent's frame ("body frame"). An application of this coordinated motion is outlined in [76] to control a fleet of underwater gliders for ocean exploration. Motion on $SE(n)$ with steering control is also directly linked to the evolution of a Serret-Frenet frame with curvature control, as explained in [60]. The general framework proposed here for coordinated

motion on Lie groups appears to be original, although [63] on $SE(2)$ and [64, 123, 126] on $SE(3)$ make explicit use of Lie group formulations at some points. Application of the present framework for appropriately underactuated left-invariant coordination, or equivalently total coordination, on $SE(2)$ leads to the control laws proposed in [130, 131, 132]; similarly, applying the framework for underactuated left-invariant coordination to $SE(3)$ actually follows the steps of [123, 126]. When translational motion is discarded, the configuration space reduces to the compact Lie group $SO(n)$, whose standard application is satellite attitude control on $SO(3)$. The latter is mostly concerned with attitude *synchronization*; see references in the introduction to Part II in this respect.

Results taking into account the full mechanical dynamics for rigid body motion are more difficult to obtain — see for instance applications of the framework of [83] for coordination on $SO(3)$ and $SE(3)$ respectively in [96, 99] and [135, 48] (just two agents, as in [64]). A generalization of this framework is not part of the present work.

A direct link can be established between the position controllers designed in the present dissertation for total coordination and the double bracket flows of [19] for gradient systems on adjoint orbits.

To conclude, it must be said that "coordinated motion" as defined in the present dissertation is certainly not the only meaningful collective motion for a swarm of agents. In [141] for instance, an organized motion is described where the relative positions of the agents change periodically over time, such that each agent "takes the lead" in turns.

Chapter 8

Definitions of "coordinated motion" and their consequences

This chapter proposes definitions for coordinated motion on Lie groups by starting from basic invariance principles. It establishes conditions on the velocities for coordination and examines their implications. Except that symmetries must be compatible, these developments are independent of the dynamics considered for the control problem. Notations, basic definitions and properties of Lie groups can be found in Section 2.4.1.

8.1 Symmetries, relative positions and coordination

Consider a swarm of N agents evolving on a Lie group G, with $g_k(t) \in G$ denoting the position of agent k at time t, for $k \in \mathcal{V} := \{1, 2, ..., N\}$.

The starting point for the following developments, in line with the fundamental approach of the whole dissertation, is to assume *invariance* or *symmetry* in the behavior of the swarm of agents with respect to their absolute position on the Lie group: only *relative* positions of the agents matter. In the three-dimensional physical world, the laws governing interactions in a set of particles are invariant with respect to translations and rotations of the whole set as a rigid body; from this viewpoint, the invariance assumption comes down to assuming that the agents are isolated, left to themselves, with no external influence directing their behavior.

The symmetries of the system impose how to define meaningful quantities for the swarm, like "relative positions" and consequently "coordinated motion". They also determine what the dynamics of the coupled agents can be: feedback control laws that asymptotically enforce coordination must be designed on the basis of error measurements only involving appropriately invariant quantities (e.g. appropriate relative positions). In Part I, the agents are assumed to behave invariantly under a uniform translation on the circle. On a Lie group, translations are implemented by group multiplication, either on the left or on the right. This leads to two possibilities to define invariant relative positions, depending on the symmetry which is considered.

Definition 8.1.1: *The* left-invariant relative position *of agent j with respect to agent k is $\lambda_{jk} = g_k^{-1} g_j$. The* right-invariant relative position *of agent j with respect to agent k is $\rho_{jk} = g_j\, g_k^{-1}$.*

Left-invariant position λ_{jk} is indeed invariant under "uniform left translation": $\forall h \in G$, $(hg_k)^{-1}(hg_j) = g_k^{-1} g_j$. Similarly, right-invariant position ρ_{jk} is invariant under "uniform right translation": $\forall h \in G$, $(g_j h)\,(g_k h)^{-1} = g_j\, g_k^{-1}$. The left-/right-invariant relative positions are the *joint invariants* associated to the left-/right-invariant action of G on the complete configuration space $G \times G ... \times G$ (N copies). In some papers (e.g. [123]), the relative positions are also called *shape variables*, because specifying all relative positions characterizes the "shape" of the formation. When a complete *setting* — i.e. the control objective and the agent dynamics — is invariant with respect to left or right multiplication, then the corresponding relative positions completely characterize the behavior of the system.

Coordinated motion is defined as motions with fixed relative positions. This is compatible with the intuition of a "formation", where a swarm of vehicles moves like a single extended "rigid body". The two different types of relative position lead to two different types of coordinated motion; by requiring the two types to hold simultaneously, a third type of coordinated motion is defined. For the sake of brevity, in the following, *coordination* is used to denote *coordinated motion.*

Definition 8.1.2: Left-invariant coordination *(LIC) denotes situations where the left-invariant relative positions $\lambda_{jk}(t) = g_k^{-1} g_j$ in the swarm are constant over time, $\forall j,\, k \in \mathcal{V}$.* Right-invariant coordination *(RIC) denotes situations where the right-invariant relative positions $\rho_{jk}(t) = g_j g_k^{-1}$ are constant over time, $\forall j,\, k \in \mathcal{V}$.*

Definition 8.1.3: Total coordination *(TC) denotes simultaneous left- and right-invariant coordination, that is situations where both left- and right-invariant positions, $\lambda_{jk}(t) = g_k^{-1} g_j$ and $\rho_{jk}(t) = g_j g_k^{-1}$, are constant over time, $\forall j,\, k \in \mathcal{V}$.*

8.2 Velocities and coordination

With coordination defined by *constant* relative positions, it is interesting to express conditions for coordination as algebraic requirements on the agents' velocities. As illustrated in Example 1.2.7, velocities on manifolds belong to different tangent spaces, so comparing velocities requires a way to translate velocity vectors from one tangent space to another. On a Lie group G, the left and right group multiplications offer two canonical ways of translating all velocity vectors to the tangent space T_eG at identity e; the resulting vectors in $T_eG = \mathfrak{g}$ are the left- and right-invariant velocities ξ^l and ξ^r respectively, see Definition 2.4.5. The link between ξ^l and ξ^r through the *adjoint representation* (see Definition 2.4.6),

$$\xi^r = Ad_g\, \xi^l \quad , \tag{8.1}$$

plays an important role in the following developments. Properties 2.4.4 and 2.4.8 are used extensively in the calculations, like for proving the following fundamental fact.

Proposition 8.2.1: *Left-invariant coordination on a Lie group G corresponds to equal right-invariant velocities $\xi_j^r = \xi_k^r\ \forall j,k \in \mathcal{V}$. Right-invariant coordination corresponds to equal left-invariant velocities $\xi_j^l = \xi_k^l\ \forall j,k \in \mathcal{V}$.*

Proof: For left-invariant coordination,

$$\begin{aligned}\tfrac{d}{dt}(g_k^{-1}g_j) &= L_{g_k^{-1}*}\tfrac{d}{dt}g_j + R_{g_j*}\tfrac{d}{dt}g_k^{-1} = L_{g_k^{-1}*}L_{g_j*}\xi_j^l - R_{g_j*}L_{g_k^{-1}*}Ad_{g_k}\xi_k^l \\ &= L_{g_k^{-1}*}(L_{g_j*}\xi_j^l - R_{g_j*}Ad_{g_k}\xi_k^l) = L_{g_k^{-1}*}L_{g_j*}(\xi_j^l - L_{g_j^{-1}*}R_{g_j*}Ad_{g_k}\xi_k^l) \\ &= L_{g_k^{-1}g_j*}(\xi_j^l - Ad_{g_j^{-1}g_k}\xi_k^l) = L_{g_k^{-1}g_j*}Ad_{g_j}^{-1}(\xi_j^r - \xi_k^r) \qquad (8.2)\end{aligned}$$

where the first equality follows from Leibniz' rule and basic rules presented around Definition 2.4.5 of velocities, the second equality uses Property 2.4.8 (c), eq. (2.5), the third equality follows from Property 2.4.4 (c), the fourth one uses Properties 2.4.4 (a) and (b), the fifth one uses Properties 2.4.4 (b) and 2.4.8 (a) in association with Definition 2.4.6 of the adjoint representation, and the last one uses Property 2.4.8 (a) again as well as the fundamental relation (8.1). Since $L_{g_k^{-1}g_j*}$ and $Ad_{g_j}^{-1}$ are invertible, requiring constant $\lambda_{jk}(t) = g_k(t)^{-1}g_j(t)$ is equivalent to the algebraic condition $\xi_j^r = \xi_k^r$.

The proof for right-invariant coordination is strictly analogous. □

The first expression in the last line of (8.2) gives the time-derivative of the left-invariant variable λ_{jk} in terms of left-invariant quantities only: velocities are left-invariant, and operators L_* and Ad are with respect to left-invariant relative positions λ_{jk} and $\lambda_{kj} = \lambda_{jk}^{-1}$.

Proposition 8.2.1 shows that coordination on the Lie group G is equivalent to consensus in the vector space $\mathfrak{g}$. This is of interest since the convergence of consensus algorithms in vector spaces is well-established in the literature under weak conditions (see Section 3.1).

According to Proposition 8.2.1, total coordination requires *simultaneous* consensus on ξ_k^l and ξ_k^r; but the latter are not independent, they are linked through (8.1) which depends on the agents' positions. This leads to the following algebraic conditions involving velocities *and positions* to ensure total coordination.

Proposition 8.2.2: *Let* $\ker(B)$ *denote the kernel of application* $B : \mathfrak{g} \to \mathfrak{g}$, *i.e.* $\ker(B) = \{\xi \in \mathfrak{g} : B\xi = 0\}$. *Total coordination on a Lie group* G *is equivalent in the Lie algebra* $\mathfrak{g}$ *to the condition* $\forall k \in \mathcal{V}$,

$$\xi_k^l = \xi^l \in \bigcap_{m,j} \ker(Ad_{\lambda_{mj}} - \text{Identity}) \quad \text{or equivalently} \quad \xi_k^r = \xi^r \in \bigcap_{m,j} \ker(Ad_{\rho_{mj}} - \text{Identity})\ .$$

Proof: RIC requires $\xi_k^l = \xi_j^l\ \forall j,k$. Denote the common value of the ξ_k^l by ξ^l. Then LIC requires $Ad_{g_k}\xi^l = Ad_{g_j}\xi^l \Leftrightarrow \xi^l = Ad_{\lambda_{jk}}\xi^l\ \forall j,k$ which is equivalent to the statement in the Proposition. The proof with ξ^r is similar. □

NB: The following discussion privileges the use of left-invariant variables; strictly similar remarks can be made using right-invariant variables.

The central requirement in Proposition 8.2.2 is

$$Ad_{\lambda_{jk}}\xi^l = \xi^l\ . \qquad (8.3)$$

It essentially says that for a fixed set of left-invariant relative positions λ_{jk}, total coordination is only achievable at left-invariant velocities ξ^l that are eigenvectors associated to eigenvalue 1 of the adjoint representation $Ad_{\lambda_{jk}}$, for all $j,\ k \in \mathcal{V}$. Conversely, for a fixed left-invariant

velocity ξ^l, total coordination can only be achieved by placing the agents at particular relative positions for which ξ^l is an eigenvector associated to eigenvalue 1 of the adjoint representation. Thus on a general Lie group, total coordination with nonzero velocity can restrict the set of possible relative positions and reciprocally total coordination with specific relative positions can restrict achievable velocities.

For an Abelian group, Ad_g = Identity $\forall g \in G$, so $\xi_k^r = \xi_k^l$ in any situation. Then RIC , LIC and TC are actually all equivalent and Proposition 8.2.2 becomes trivial.

8.2.1 Relative positions compatible with TC at a particular velocity

Proposition 8.2.3: *Let* $CM_\xi := \{g \in G : Ad_g\,\xi = \xi\}$.
(a) *For every* $\xi \in \mathfrak{g}$, CM_ξ *is a Lie subgroup of* G.
(b) *The Lie algebra of* CM_ξ *is the kernel of* $ad_\xi := [\xi,]$, *i.e.* $\mathfrak{cm}_\xi = \{\eta \in \mathfrak{g} : [\xi, \eta] = 0\}$.
(c) *The adjoint orbit* $O_\xi = \{Ad_g\xi : g \in G\}$ *of* ξ *is the quotient of* G *by its subgroup* CM_ξ.

Proof: (a) $Ad_e\,\xi = \xi\ \forall\xi$ since Ad_e is the identity operator. $Ad_g\,\xi = \xi$ implies $Ad_{g^{-1}}\,\xi = Ad_{g^{-1}}\,Ad_g\,\xi = \xi$. Moreover, if $Ad_{g_1}\,\xi = \xi$ and $Ad_{g_2}\,\xi = \xi$, then $Ad_{g_1g_2}\,\xi = Ad_{g_1}\,Ad_{g_2}\,\xi = Ad_{g_1}\,\xi = \xi$. Thus CM_ξ satisfies all group axioms and must be a subgroup of G. Then Cartan's theorem, recalled in Proposition 2.4.3, ensures that CM_ξ is a *Lie* subgroup of G if it is closed under the topology of G; this holds by continuity of $Ad_g\,\xi$ in g and in ξ.

(b) Let $g(t) \in CM_\xi$ with $g(\tau) = e$ and $\frac{d}{dt}g(t)|_\tau = \eta$. Then $\eta \in \mathfrak{cm}_\xi$ = the tangent space to CM_ξ at e. For constant ξ, $Ad_g(t)\xi = \xi$ implies $\frac{d}{dt}(Ad_g(t))\xi = 0$, which at $t = \tau$, thanks to Property 2.4.8 (c), eq. (2.6), means $[\eta, \xi] = 0$ necessarily. It is also sufficient since, for any η such that $[\eta, \xi] = 0$, the group exponential curve $g(t) = \exp(\eta t)$ belongs to CM_ξ.

(c) The elements of O_ξ write $Ad_g\,\xi$ with $g \in G$. Define equivalence classes in G such that g_1 and g_2 are equivalent if and only if $g_1^{-1}g_2 \in CM_\xi$, i.e. $Ad_{g_1^{-1}g_2}\,\xi = \xi$ or equivalently $Ad_{g_1}\,\xi = Ad_{g_2}\,\xi$; this makes g_1 and g_2 equivalent if and only if they correspond to the same point of O_ξ. □

Definition 8.2.4: CM_ξ *and* $\mathfrak{cm}_\xi$ *are the* isotropy subgroup *and* isotropy algebra *of* ξ. *Relative positions* λ_{jk} *belonging to* CM_ξ *are said to be* compatible *with total coordination at velocity* ξ.

The isotropy subgroup and its Lie algebra are classical objects in group theory [83]; thus the sets of relative positions compatible with total coordination at particular velocities can be considered as well characterized. In particular, one method to obtain a totally coordinated motion on Lie group G is to
(1) impose $\xi_k^l = \xi^l\ \forall k \in \mathcal{V}$ with ξ^l chosen in the vector space $\mathfrak{g}$ and
(2) position the agents such that $\lambda_{jk} \in CM_{\xi^l}$ for a set of edges (j, k) forming a weakly connected digraph.
Step (1) ensures RIC , and Proposition 8.2.3 (a) ensures that after Step (2), *all* relative positions — not only those corresponding to the edges selected in Step (2) — are compatible with total coordination at velocity ξ^l. For the choice $\xi^l = 0$, any set of relative positions is compatible, i.e. $CM_0 = G$; but this is not really interesting for coordinated "motion".

8.2.2 Velocities compatible with TC and particular relative positions

Another method to obtain a totally coordinated motion on a Lie group G is to

(1) choose initial positions $g_k(0)$ for the agents on G and
(2) impose velocity $\xi_k^l = \xi^l \; \forall k \in \mathcal{V}$ where ξ^l satisfies $Ad_{\lambda_{mj}}\xi^l = \xi^l$ for all pairs of agents (m, j) corresponding to the edges of a weakly connected digraph.

Proposition 8.2.3 (a) again ensures that considering pairs of agents corresponding to the edges of a weakly connected graph is sufficient to ensure compatibility for *all* pairs of agents. Because relative positions are imposed before the velocity, ξ^l is now said to be *compatible* with TC and specific relative positions if it satisfies the appropriate conditions of Proposition 8.2.2. For later reference, let $\Lambda(t)$ (resp. $P(t)$) denote the set of all relative positions λ_{jk} (resp. ρ_{jk}) for $j, k \in \mathcal{V}$, and define

$$SP_1(\Lambda) = \bigcap_{j,k} \ker(Ad_{\lambda_{jk}} - \text{Identity}) \quad \text{for the } \lambda_{jk} \text{ of } \Lambda\,. \tag{8.4}$$

With this notation, the condition for the velocity writes $\xi_k^l = \xi^l \in SP_1(\Lambda(0)) \; \forall k \in \mathcal{V}$.

The velocities compatible with a set of relative positions are computed as intersections of eigenspaces of a set of linear operators, so they can be considered as well characterized. In particular, for a fixed set of relative positions, the set of compatible velocities is always a vector subspace of $\mathfrak{g}$. However, since this subspace results from an intersection of subspaces corresponding to $N-1$ independent relative positions, it quickly reduces to the singleton $\{0\}$ for a sufficient number N of agents with generic positions on non-Abelian groups. For agents synchronized at the same position, any velocity is compatible with TC , i.e. $SP_1(e, e, ..., e) = \mathfrak{g}$; this completely degenerate case is not really insightful, it could be proposed on any manifold without a general theory.

8.2.3 Changing velocities and relative positions in coordinated motion

For constant ξ_k^l, Property 2.4.8 (c), eq. (2.6) implies that ξ_k^r is constant as well. Thus if $\frac{d}{dt}\xi_k^l = 0 \; \forall k \in \mathcal{V}$ and $\xi_k^r(0) = \xi_j^r(0) \; \forall j, k \in \mathcal{V}$, then $\xi_k^r(t) = \xi_j^r(t)$ for all t. In words, according to Proposition 8.2.1, if LIC is achieved initially and left-invariant velocities are kept constant, then LIC is maintained $\forall t$. More obviously, if RIC is achieved initially and left-invariant velocities are kept constant, then RIC is maintained $\forall t$. Combining the two previous sentences, if TC is achieved initially and left-invariant velocities are kept constant, then TC is maintained $\forall t$. The next paragraphs examine what can be done by *varying velocities* $\xi^l(t)$. The following definition is needed.

Definition 8.2.5: *A group element $g_1 \in G$ is a* conjugate *of element $g_2 \in G$ if there exists $h \in G$ such that $g_1 = h\, g_2\, h^{-1}$; elements g_1 and g_2 are then said to be* conjugated by h. *The set of all elements that are conjugates of some element $g \in G$ forms a* conjugacy class.

Proposition 8.2.6: *The subspace of T_gG tangent at $g \in G$ to the conjugacy class of g is $\{L_{g*}(Ad_g - \text{Identity})\xi : \xi \in \mathfrak{g}\}$. As a consequence, the dimension of the conjugacy class of g is equal to* $\text{dimension}(G) - \text{dimension}(SP_1(g))$.

Proof: Write $g_c(t) = h(t)^{-1}\, g\, h(t)$ with g fixed and consider all possible trajectories of $h(t)$ with $h(\tau) = e$. The tangent space at g to the conjugacy class of g contains all vectors that can be associated to $\frac{d}{dt}g_c|_\tau$. Denoting by ξ_h^l the left-invariant velocity of $h(t)$ at $t = \tau$ yields

$$\tfrac{d}{dt}g_c|_\tau = L_{h(\tau)^{-1}gh(\tau)*}\,\xi_h^l - R_{h(\tau)^{-1}gh(\tau)*}\,\xi_h^l = L_{g*}\xi_h^l - R_{g*}\xi_h^l = L_{g*}(\xi_h^l - Ad_g^{-1}\xi_h^l)$$

where ξ_h^l can take any value in $\mathfrak{g}$. The change of variables $\xi = Ad_g^{-1}\xi_h^l$ yields the form stated in the Proposition. The linear operator $L_{g*} : \mathfrak{g} \to \mathfrak{g}$ is invertible, thus it has full rank. By

definition, the dimension of the kernel of $(Ad_g - \text{Identity})$ is dimension$(SP_1(g))$, such that its image has dimension(G) − dimension$(SP_1(g))$. This concludes the proof since the dimension of a manifold is equal to the dimension of its tangent space. □

Proposition 8.2.6 highlights a direct link between the dimension of the set $SP_1(g)$ of velocities compatible with TC and relative position g, and the dimension of the conjugacy class of g. The latter is the set of relative positions accessible in a coordinated way from initial relative position g, as shown in the following.

Assume that a swarm is in an original state of total coordination characterized by relative positions $\Lambda(0)$, $P(0)$ and velocities $\xi^l(0) \in SP_1(\Lambda(0))$, $\xi^r(0) \in SP_1(P(0))$. The question is which other (final) states, characterized by relative positions $\Lambda(T)$, $P(T)$ and velocities $\xi^l(T)$, $\xi^r(T)$, can be reached from this original state *in a coordinated way.*

1. If total coordination is maintained during the transition from 0 to T, then the λ_{jk} and ρ_{jk} do not change such that $\Lambda(T) = \Lambda(0)$, $P(T) = P(0)$. The velocities may freely vary in the original eigenspace intersection (as long as they remain equal $\forall k \in \mathcal{V}$): $\xi^l(T) \in SP_1(\Lambda(0))$, $\xi^r(T) \in SP_1(P(0))$.

2. If right-invariant coordination is maintained during the transition from 0 to T, then $\xi^l_k(t) = \xi^l_j(t)$ $\forall j,k \in \mathcal{V}$ throughout the motion (Proposition 8.2.1).

   First consider the relative positions which are "accessible" from $\Lambda(0)$, $P(0)$ by maintainig right-invariant coordination. Moving with equal ξ^l_k implies that $g_k(T) = g_k(0)h$ $\forall k \in \mathcal{V}$, for some fixed $h \in G$. This is in agreement with Definition 8.1.2 of RIC as constant ρ_{jk}, implying $\rho_{jk}(T) = \rho_{jk}(0)$ $\forall j,k$ such that $P(T) = P(0)$. It also implies $\lambda_{jk}(T) = h^{-1}\lambda_{jk}(0)\,h$ $\forall j,k$, meaning that all $\lambda_{jk}(T)$ are conjugated by h with respect to $\lambda_{jk}(0)$. Thus *the set of accessible* $\Lambda(T)$ *is the conjugacy class of* $\Lambda(0)$ (where conjugation acts componentwise, but with the same element h, on the elements λ_{jk} of Λ).

   Next consider which velocities *compatible with total coordination* at time T are accessible. Like in the previous case, $P(T) = P(0)$ implies $\xi^r(0) \in SP_1(P(0))$ such that right-invariant velocities can only evolve in the original eigenspace intersection. For the left-invariant velocities, condition (8.3) becomes $Ad_{h^{-1}\,\lambda_{jk}(0)\,h}\,\xi^l(T) = \xi^l(T)$ or equivalently

   $$Ad_{\lambda_{jk}(0)}\,(Ad_h\,\xi^l(T)) = (Ad_h\,\xi^l(T)) \quad \forall j,k \in \mathcal{V}\,.$$

   This means that $Ad_h\,\xi^l(T)$ must belong to $SP_1(\Lambda(0))$ for some h, or equivalently that *the accessible* $\xi^l(T)$ *belong to the adjoint orbit* $O_{SP_1(\Lambda(0))}$ *of* $SP_1(\Lambda(0))$ — where Definition 2.4.9 of adjoint orbits is generalized to subsets $\mathcal{S}$ of $\mathfrak{g}$ to mean $O_{\mathcal{S}} = \{Ad_g\xi : g \in G \text{ and } \xi \in \mathcal{S}\}$.

3. Similarly, if left-invariant coordination is maintained during the transition, then $\Lambda(T) = \Lambda(0)$ and the accessible $P(T)$ belong to the conjugacy class of $P(0)$. For velocities compatible with *total coordination* at time T, $\xi^l(T) \in SP_1(\Lambda(0))$ and the accessible $\xi^r(T)$ belong to the adjoint orbit $O_{SP_1(P(0))}$ of $SP_1(P(0))$.

The adjoint orbit and the partition of G into conjugacy classes are classical tools of group theory; thus the above possibilities with varying velocities are well characterized.

8.2.4 Using total coordination to define specific configurations

A *configuration* is a set of particular values of the agents' relative positions (see Definition 3.4.1). In many applications involving coordinated motion, reaching a particular configuration is a relevant problem. Part II of the present dissertation defines some specific configurations — namely: synchronization, consensus, anti-consensus and balancing — for agents evolving on compact Lie groups (and homogeneous manifolds). The fact that achieving TC at a particular velocity ξ restricts the set of compatible relative positions might be used as an indirect way to specify particular formations. One advantage of defining configurations through TC compatibility is that, unlike Part II, the present setting is not reduced to *compact* manifolds.

The set of configurations compatible with TC at a fixed velocity ξ forms a continuum, because the set CM_ξ to which relative positions must belong is a Lie subgroup of G. In order to get more restricted sets of configurations, one could require compatibility with TC at several different velocities $\xi_1, \xi_2, ..., \xi_p$: then the relative positions would have to belong to the intersection of isotropy groups $\bigcap_{k=1}^{p} CM_{\xi_k}$. Still more variants might be obtained by allowing relative positions to belong to the union of such intersections, $\bigcup_{j=1}^{m} \left(\bigcap_{k=1}^{p} CM_{\xi_{kj}}\right)$. These ideas have not been really explored yet.

O.Q.: It is currently unknown to which extent the restriction imposed on relative positions by total coordination at a fixed velocity can be used to indirectly define specific (classes of) configurations that could be useful in practice.

8.2.5 Examples

The following examples illustrate the proposed theory on three particular groups $SO(3)$, $SE(2)$ and $SE(3)$. These groups are chosen mainly for their importance in classical engineering applications as characterizing rigid bodies moving in the physical world; the group $SO(2)$ of orientations in the plane is Abelian, and therefore the problem of *motion coordination* — unlike position coordination — trivially reduces to agreement in a vector space, namely on a rotation rate $\omega \in \mathbb{R}$. At the beginning of each example, the group properties presented in Section 2.4 for $SO(n)$ and $SE(n)$ are recalled and particularized to the specific dimension. An interpretation of vehicle motion on the sphere in terms of the group $SO(3)$ is also provided as an illustration of the general applicability of the framework. Left-invariant coordination for the particular examples of $SE(2)$ and $SE(3)$ was already formulated in Lie group notation in [63, 64].

Ex. 8.2.7: coordinated motion on $SO(3)$: A point g on $SO(3)$ is represented by a 3-dimensional rotation matrix Q and can be thought of as representing the orientation of a 3-dimensional rigid body with respect to an arbitrary inertial frame.

- Group multiplication, inverse and identity are the corresponding matrix operations.
- The Lie algebra $\mathfrak{so}(3)$ is the set of antisymmetric 3×3 matrices $[\omega]^\wedge$; operations $L_{Q*}\xi$ and $R_{Q*}\xi$ are represented by $Q[\omega]^\wedge$ and $[\omega]^\wedge Q$ respectively. Notation $[\omega]^\wedge$ is used because several interesting properties follow from identifying $\mathfrak{so}(3) \ni [\omega]^\wedge$ with $\mathbb{R}^3 \ni \omega$ through the invertible mapping

$$\begin{pmatrix} 0 & -\omega_{(3)} & \omega_{(2)} \\ \omega_{(3)} & 0 & -\omega_{(1)} \\ -\omega_{(2)} & \omega_{(1)} & 0 \end{pmatrix} \in \mathfrak{so}(3) \qquad \underset{[\cdot]^\wedge}{\overset{[\cdot]^\vee}{\rightleftarrows}} \qquad \begin{pmatrix} \omega_{(1)} \\ \omega_{(2)} \\ \omega_{(3)} \end{pmatrix} \in \mathbb{R}^3 \quad .$$

A fundamental property is $[a]^\wedge b = a \wedge b \ \ \forall a$ and $b \in \mathbb{R}^3$, where $\wedge$ is the vector product.

- With this identification, $Ad_Q\omega = Q\omega$ and $[\omega_k, \omega_j] = [\omega_k]^\wedge\omega_j = \omega_k \wedge \omega_j$.
- In the standard interpretation of Q as rigid body orientation, ω^l and ω^r are the angular velocities expressed in body frame and in inertial frame respectively.

Thanks to the interpretation in the last item, LIC (equal ω_k^r), RIC (equal ω_k^l) and TC have a clear physical meaning, see Figure 8.1. Straightforward computations allow to characterize the interplay between velocities and positions for TC as follows.

For a fixed velocity $\omega \neq 0$, compatible relative positions are characterized by $\mathfrak{cm}_\omega = \{\lambda\omega : \lambda \in \mathbb{R}\}$ and $CM_\omega = \{\text{rotations around axis } \omega\}$. The dimension of $\mathfrak{cm}_{\xi^l}$ ($\Leftrightarrow$ of CM_{ξ^l}) is 1; the agents rotate with the same angular velocity ω_k^r in inertial frame and must have the same orientation up to a rotation around ω_k^r, see the drawing on the right in Figure 8.1.

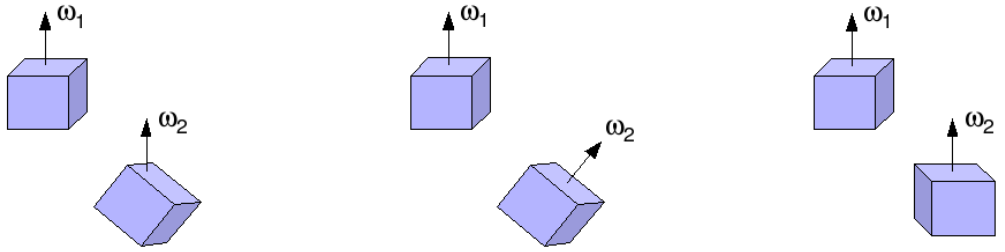


Figure 8.1: Illustration of left-invariant (left), right-invariant (middle) and total coordination (right) of two agents on $SO(3)$. The arrows represent angular velocities, viewed as ω_k^r by the reader attached to its inertial frame.

For a fixed relative position $Q_k^T Q_j \neq I_3$, the eigenspace of eigenvalue 1 of $Ad_{Q_k^T Q_j}$ is $SP_1(Q_k^T Q_j) = \{\lambda\alpha : \lambda \in \mathbb{R} \text{ and } \alpha \text{ is the Euler rotation axis between } Q_k \text{ and } Q_j\}$. For generic positions, the intersection of such eigenspaces just leaves $SP_1(\Lambda) = \{0\}$ as soon as $N > 2$.

Regarding positions and velocities that are reachable in a coordinated way,

- the adjoint orbit of ω is $O_\omega = \{\lambda \in \mathbb{R}^3 : \|\lambda\| = \|\omega\|\}$, so the adjoint orbit of any vector space (different from a singleton) is the whole set of angular velocities $\mathbb{R}^3 \cong \mathfrak{so}(3)$;
- the conjugacy class of $Q_k \in SO(3)$ is the set of elements $Q_j \in SO(3)$ that are at the same *chordal distance* of the identity, i.e. (see Chapter 5) which satisfy $\text{trace}(Q_k) = \text{trace}(Q_j)$. Indeed, this condition is necessary since $\text{trace}(QQ_kQ^T) = \text{trace}(Q_kQ^TQ) = \text{trace}(Q_k)$; it is also sufficient essentially because the dimension of the set of rotation matrices at the same distance of the identity equals 2 for $Q_k \neq I_3$ and 0 for $Q_k = I_3$, exactly like the dimension of the conjugacy class expected according to Proposition 8.2.6.

◇

Ex. 8.2.8: coordinated motion on $SE(2)$: The special Euclidean group in the plane $SE(2)$ describes all planar rigid body motions (translations and rotations). An element of $SE(2)$ can be written $g = (r, \theta)$ where $r \in \mathbb{R}^2$ denotes position and $\theta \in S^1$ denotes orientation.

- Group multiplication $g_1g_2 = (r_1 + Q_{\theta_1}r_2, \theta_1 + \theta_2)$ where Q_θ is the planar rotation of angle θ. Identity $e = (0, 0)$ and inverse $g^{-1} = (-Q_{-\theta}r, -\theta)$.
- Lie algebra $\mathfrak{se}(2) = \mathbb{R}^2 \ltimes \mathbb{R} \ni \xi = (v, \omega)$. Operations $L_{g*}(v, \omega) = (Q_\theta v, \omega)$ and $R_{g*}(v, \omega) = (v + \omega Q_{\pi/2}r, \omega)$.
- $Ad_g(v, \omega) = (Q_\theta v - \omega Q_{\pi/2}r, \omega)$ and $[(v_1, \omega_1), (v_2, \omega_2)] = (\omega_1 Q_{\pi/2}v_2 - \omega_2 Q_{\pi/2}v_1, 0)$.

- In the interpretation of rigid body motion, v^l is the body's linear velocity expressed in body frame, $\omega^l = \omega^r =: \omega$ is its rotation rate. However, for $\omega \neq 0$, v^r is not the body's linear velocity expressed in inertial frame. Instead, $s = \frac{-Q_{\pi/2}}{\omega} v^r$ is the center of the circle drawn by the rigid body moving with $\xi^r = (v^r, \omega)$. In [131], the intuitive argument for achieving coordination is to synchronize circle centers s_k; this actually means synchronizing right-invariant velocities v_k^r ($\neq$ linear velocities expressed in inertial frame).

In RIC , the agents move with the same velocity expressed in body frame, such that they "draw similar trajectories" (see Figure 8.2.r). In LIC , they move like a single rigid body: relative orientations and relative positions on the plane do not change (see Figure 8.2.l_1 and 8.2.l_2). In TC , the swarm moves like a single rigid body *and* each agent has the same velocity expressed in body frame. The associated conditions on velocities and relative positions can be characterized as follows.

For a fixed velocity ξ^l, Propositions 8.2.2 and 8.2.3 characterize $\mathfrak{cm}_{\xi^l}$ by $[\xi^l, \eta] = 0 \Leftrightarrow \omega^l v_\eta = \omega_\eta v^l$ and CM_{ξ^l} by $Ad_g\, \xi^l = \xi^l \Leftrightarrow (Q_\theta - I_2) v^l = \omega^l Q_{\pi/2} r$. This leads to two different cases with $\xi^l \neq 0$.

1. (Parallel motion) If $\omega^l = 0$ and $v^l \neq 0$, then $\mathfrak{cm}_{\xi^l} = \{(\lambda, 0) : \lambda \in \mathbb{R}^2\}$ and $CM_{\xi^l} = \{(r, 0) : r \in \mathbb{R}^2\}$; the dimension of $\mathfrak{cm}_{\xi^l}$ ($\Leftrightarrow$ of CM_{ξ^l}) is 2. The agents have the same orientation and move on parallel straight lines (see Figure 8.2.t_1).
2. (Circular motion) If $\omega^l \neq 0$ and v^l is arbitrary, then $\mathfrak{cm}_{\xi^l} = \{(\frac{\lambda}{\omega^l} v^l, \lambda) : \lambda \in \mathbb{R}\}$. Define C, the circle of radius $\frac{\|v^l\|}{|\omega^l|}$ containing the origin, tangent to v^l at the origin and such that v^l and ω^l imply rotation in the same direction. Then solving $Ad_g \xi^l = \xi^l$ for g and making a few calculations shows that $CM_{\xi^l} = \{(r, \theta) : r \in C$ and $Q_\theta v^l$ tangent to C at r implying rotation in the same direction as ω^l $\}$. The dimension of $\mathfrak{cm}_{\xi^l}$ ($\Leftrightarrow$ of CM_{ξ^l}) is 1. The agents move on the same circle and have the same orientation with respect to their local radius (see Figure 8.2.t_2).

To characterize velocities compatible with TC at a fixed relative position $g_k^{-1} g_j \neq e$, the same two cases can be distinguished, in addition to a further degenerate case.

1. If $\theta_j = \theta_k$, then $SP_1(g_k^{-1} g_j) = \{(v, 0) \in \mathfrak{se}(2) : v \in \mathbb{R}^2\}$, i.e. angular velocity must be zero and linear velocity can be arbitrary.
2. If $\theta_j \neq \theta_k$ and $r_j \neq r_k$, then there exists a unique circle C, of finite radius r and center c, which contains both positions r_k and r_j and such that $\arg(r_k - c) - \arg(r_j - c) = \theta_k - \theta_j$ (identifying the plane with $\mathbb{C}$ to define arg as in Parts I and II). Then $SP_1(g_k^{-1} g_j) = \{(v, \frac{\varepsilon \|v\|}{r}) : v$ is tangent to C and $\varepsilon = +1$ (resp. -1) if v implies clockwise (resp. counterclockwise) motion around $C\}$.
3. If $\theta_k \neq \theta_j$ and $r_j = r_k$, then $SP_1(g_k^{-1} g_j) = \{(0, \lambda) : \lambda \in \mathbb{R}\}$.

The second case is the generic one, such that for generic positions, the intersection of eigenspaces just leaves $SP_1(\Lambda) = \{0\}$ as soon as $N > 2$.

The adjoint orbits and conjugacy classes can be computed as an exercise; again the two cases $\omega = 0$ and $\omega \neq 0$ must be distinguished. An interesting remark in this context is that unless $g_1 = g_2 = ...g_N$, an initial situation compatible with TC circular motion can never transform into a situation compatible with TC parallel motion in a coordinated way, and reciprocally; indeed, because $\omega_k = \omega_j \ \forall j, k$ for any type of coordination, relative orientations will never be able to change. $\diamond$

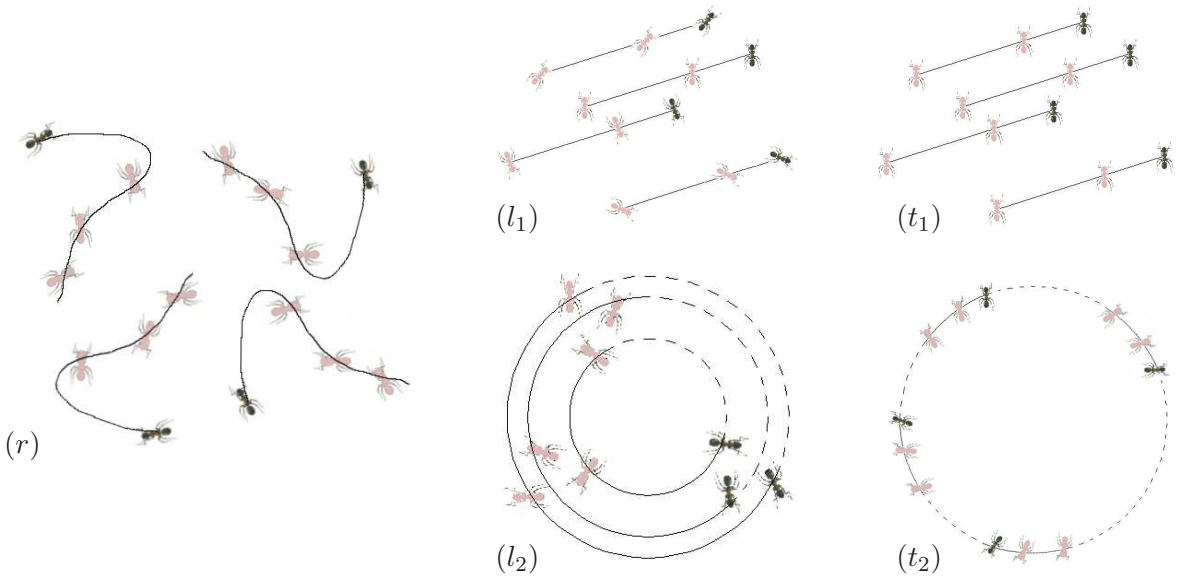


Figure 8.2: Coordinated swarms in $SE(2)$. (r): RIC with varying velocity. (l_1) and (l_2): LIC with $\omega_k = 0$ and $\omega_k \neq 0$ respectively. (t_1) and (t_2): TC with $\omega_k = 0$ and $\omega_k \neq 0$ respectively. Intermediate (in time) positions and orientations of the agents are represented in light color.

Ex. 8.2.9: coordinated motion on $SE(3)$: The group $SE(3)$ describes all 3-dimensional rigid body motions (translations and rotations). An element of $SE(3)$ can be written $g = (r, Q)$, with $r \in \mathbb{R}^3$ a vector characterizing the body's position and $Q \in SO(3)$ a rotation matrix characterizing its orientation.

- Group multiplication $g_1 g_2 = (r_1 + Q_1 r_2, Q_1 Q_2)$, identity $e = (0, I_3)$ and inverse $g^{-1} = (-Q^T r, Q^T)$.
- Lie algebra $\mathfrak{se}(3) = \mathbb{R}^3 \ltimes \mathfrak{so}(3) \ni \xi = (v, [\omega]^\wedge)$ is identified with $\mathbb{R}^3 \ltimes \mathbb{R}^3 \ni (v, \omega)$ with the same mapping as for $SO(3)$. Operations $L_{g*}(v, [\omega]^\wedge) = (Qv, Q[\omega]^\wedge)$ and $R_{g*}(v, [\omega]^\wedge) = (\omega \wedge r + v, [\omega]^\wedge Q)$.
- $Ad_g(v, \omega) = (Qv + r \wedge (Q\omega), Q\omega)$ and $[(v_1, \omega_1), (v_2, \omega_2)] = (\omega_1 \wedge v_2 - \omega_2 \wedge v_1, \omega_1 \wedge \omega_2)$.
- In the interpretation of rigid body motion, left-invariant velocities v^l and ω^l are the body's linear and angular velocity respectively, expressed in body frame. Velocity ω^r is the angular velocity expressed in inertial frame. For $\omega^l \neq 0$, there is no intuitive physical interpretation for v^r.

Similarly to $SE(2)$, in RIC the agents move with the same velocity expressed in body frame, such that they draw "similar trajectories", i.e. translated and rotated versions of the same curve. In LIC they move with fixed relative orientations and relative positions, like a single 3-dimensional rigid body. In TC , the swarm moves like a single rigid body *and* the agents have identical velocities expressed in body frame. The associated conditions on velocities and relative positions can be characterized as follows.

For a fixed velocity ξ^l, Propositions 8.2.2 and 8.2.3 characterize $\mathfrak{cm}_{\xi^l}$ by $[\xi^l, \eta] = 0 \Leftrightarrow \omega^l \wedge \omega_\eta = 0$ and $\omega^l \wedge v_\eta = \omega_\eta \wedge v^l$; CM_{ξ^l} requires $Ad_g \xi^l = \xi^l \Leftrightarrow \ Q\omega^l = \omega^l$ and $(Q - I_3)v^l = \omega^l \wedge r$. Two cases can be distinguished with $\xi^l \neq 0$, similarly to $SE(2)$. The results obtained by solving for g in $Ad_g \xi^l = \xi^l$ and making several basic computations are less obvious to find

intuitively than for $SE(2)$.

1. (Parallel motion) If $\omega^l = 0$ and $v^l \neq 0$, then $\mathfrak{cm}_{\xi^l} = \{(\beta, \alpha v^l) : \beta \in \mathbb{R}^3,\ \alpha \in \mathbb{R}\}$ and $CM_{\xi^l} = \{(r,\, Q) : r \in \mathbb{R}^3$ and Q characterizes rotation around axis $v^l\}$. The dimension of $\mathfrak{cm}_{\xi^l}$ ($\Leftrightarrow$ of CM_{ξ^l}) is 4, corresponding to arbitrary positions and one rotational degree of freedom. The agents move on parallel straight lines and have the same orientation up to rotation around their linear velocity vector.
2. (Circular and Helicoidal motion) If $\omega^l \neq 0$ and v^l is arbitrary, then $\mathfrak{cm}_{\xi^l} = \{(\alpha v^l + \beta\omega^l, \alpha\omega^l) : \alpha,\, \beta \in \mathbb{R}\}$ and $CM_{\xi^l} = \{(r,\, Q) \in SE(3)$ describing left-invariant relative positions of agents that are on the same cylinder of axis ω^l and radius $\frac{\|v^l - \omega^l\,(\omega^l)^T(v^l)/\|\omega^l\|^2\|}{\|\omega^l\|}$, with orientations differing by rotations around axis ω^l through an angle exactly equal to their relative angular position on the cylinder $\}$. The dimension of $\mathfrak{cm}_{\xi^l}$ ($\Leftrightarrow$ of CM_{ξ^l}) is 2, corresponding to the dimension of the cylinder's surface. For $v^l - \omega^l\,(\omega^l)^T(v^l)/\|\omega^l\|^2 \neq 0$, the agents draw helices of constant *pitch* $(v^l)^T\omega^l = (v^r)^T\omega^r$ on the cylinder. In the special case $(v^l)^T\omega^l = 0$, the trajectories become circles in parallel planes that are perpendicular to the cylinder axis. In the degenerate situation $v^l - \omega^l\,(\omega^l)^T(v^l)/\|\omega^l\|^2 = 0$, all agents are on the rotation axis.

The same two cases are distinguished to characterize velocities compatible with TC and a fixed relative position $g_k^{-1}g_j \neq e$, in addition to a third degenerate case. Denote by α_{jk} the expression in inertial frame of the rotation axis between Q_j and Q_k.

1. If $Q_k = Q_j$, then $SP_1(g_k^{-1}g_j) = \{(v, \lambda Q_k^T(r_j - r_k)) : v \in \mathbb{R}^3$ and $\lambda \in \mathbb{R}\}$: linear velocity can be arbitrary and angular velocity must be parallel to the relative position.
2. If $Q_k \neq Q_j$ and $(r_j - r_k) \wedge \alpha_{jk} \neq 0$, then $SP_1(g_k^{-1}g_j) = \{(\lambda_2 Q_k^T\alpha_{jk} + \lambda_1(Q_k^TQ_j - I_3)^\dagger\, Q_k^T\,(\alpha_{jk} \wedge (r_j - r_k)),\ \lambda_1 Q_k^T\alpha_{jk}) : \lambda_1 \in \mathbb{R}$ and $\lambda_2 \in \mathbb{R}\}$ where $\dagger$ denotes the pseudo-inverse, which generalizes the inverse to non-invertible matrices by assigning inverse 0 to the directions in their kernel. This somewhat complex expression for SP_1 intuitively means that there exists one cylinder parallel to α_{jk} containing both agents, on which SP_1 forces them to move with appropriate velocities.
3. If $Q_k \neq Q_j$ and $(r_j - r_k) \wedge \alpha_{jk} = 0$, then $SP_1(g_k^{-1}g_j) = \{(\lambda_1 Q_k^T\alpha_{jk}, \lambda_2 Q_k^T\alpha_{jk}) : \lambda_1 \in \mathbb{R}$ and $\lambda_2 \in \mathbb{R}\}$. Then the linear velocity in inertial frame is along α_{jk}, the unique direction in $\mathbb{R}^3$ for which $Q_k^T\alpha_{jk} = Q_j^T\alpha_{jk}$, and rotation is around this same direction.

The second case is the generic one. As for $SE(2)$ and $SO(3)$, with generic positions, $SP_1(\Lambda)$ reduces to $\{0\}$ as soon as $N > 2$. Indeed, two independent relative orientations would require two different axes α for the cylinder.

The particularization of adjoint orbits and conjugacy classes for the different cases on $SE(3)$ is not considered in the present work. $\diamond$

Ex. 8.2.10: vehicles on the sphere as coordinated motion on $SO(3)$: Vehicles restricted to move on the surface of a planet for instance are characterized by a position on the sphere S^2 and a heading in the local plane of the agent, see Figure 1.4 in the general Introduction. Coordinated motion in this context should be intuitively defined equivalently to 3-dimensional rigid body coordination, involving $SE(3)$ as in the previous example, but with restricted motion of the agents: the left-invariant velocities ξ_k^l are restricted to a set ensuring that the agents stay on the sphere and keep their body vertical axis aligned with the local vertical on the sphere (e.g. the vehicles always keep the wheels on the ground). In this case, the motion actually takes place on a subgroup of $SE(3)$. Thus the method proposed in these chapters can be applied directly to the subgroup. The particular subgroup for vehicle

motion on the sphere is in fact $SO(3)$.

Indeed, just consider the orientation Q_k of a vehicle in 3-dimensional space. Assume without loss of generality that the first column-vector $\mathbf{e}_{k\,1}$ of Q_k gives the direction in inertial space from the bottom to the top of the vehicle. Then $\mathbf{e}_{k\,1}$ also fixes the position of agent k on the sphere, since the vector from the center of the sphere to its surface is exactly aligned with the vector from the bottom to the top of the vehicle. The remaining part of the orientation matrix specifies the orientation around $\mathbf{e}_{k\,1}$, which exactly corresponds to the heading of the vehicle in its local plane. In the same way, an angular velocity ω_k^l or ω_k^r for the orientation of the vehicle directly implies an associated translational velocity on the sphere.

LIC on $SO(3)$ corresponds to motion of the vehicles "in formation", i.e. with constant relative positions and orientations in three dimensions, and RIC on $SO(3)$ corresponds to drawing similar trajectories on the sphere, as for the previous examples on $SE(2)$ and $SE(3)$. According to Example 8.2.7, TC at velocity ω^r on $SO(3)$ requires body orientations to be equal up to a rotation around axis ω^r. For positions on the sphere, this implies that the agents must all be on the same parallel of the sphere with respect to axis ω^r.

This example "corrects" the discussion in Example 1.2.7 of the Introduction, where agent headings were not explicitly taken into account. ◇

Chapter 9

Designing control laws to stabilize coordinated motion

Left-invariant[1] systems on Lie groups naturally appear in many physical systems, such as rigid bodies in space and cart-like vehicles. Motivated by examples like two-axes attitude control or steering control on $SE(2)$ or $SE(3)$, this dissertation considers left-invariant dynamics with affine control of the type

$$\frac{d}{dt} g_k = L_{g_k *}\, \xi_k^l \qquad \text{with} \qquad \xi_k^l = a + B u_k \qquad , \quad k = 1, 2, ..., N \,, \tag{9.1}$$

where the Lie algebra $\mathfrak{g}$ is identified with $\mathbb{R}^n$, $a \in \mathbb{R}^n$ is a constant drift velocity, $B \in \mathbb{R}^{n \times m}$ has full column rank and specifies the range of the control term $u_k \in \mathbb{R}^m$. The set of all assignable ξ_k^l is denoted $\mathcal{C} = \{a + Bu : u \in \mathbb{R}^m\}$. For fully actuated agents $m = n$, (9.1) boils down to $\frac{d}{dt} g_k = L_{g_k *}\, u_k$ with $u_k \in \mathfrak{g}$. Feedback control laws must be functions of variables which are compatible with the symmetry of the problem setting. Therefore in the present setting, the control inputs u_k must be designed on the basis of left-invariant variables. Expressing the control objectives in terms of left-invariant variables as well, LIC corresponds to fixed (left-invariant) relative positions, while RIC corresponds to equal (left-invariant) velocities.

As in Parts I and II, the main issue of the control problem is for the agents in the swarm to reach an *agreement* on their behavior, in the absence of external reference. The information flow between agents is modeled by restricting communication links to the edges of an unweighted graph $\mathbb{G}$, where $j \rightsquigarrow k$ denotes that k gets information about j.

On Lie groups, the rank condition in Proposition 2.3.3 (Jurdjevic-Quinn theorem) becomes the following: at each point, taking consecutive Lie brackets of the drift vector a with control vectors must span the whole Lie algebra $\mathfrak{g}$. This is a necessary condition to establish controllability of invariant systems on Lie groups; for the ubiquitous examples $SO(n)$ and $SE(n)$, it is also sufficient for controllability (see [60]). Therefore, to apply these results in the following it is sufficient to check controllability with the algebraic tools of [60].

[1] This is a convention: right-invariant systems are equivalent, just by redefining group multiplication.

9.1 Coordination by consensus in the Lie algebra

9.1.1 Right-invariant coordination

Right-invariant coordination requires $\xi_k^l = \xi_j^l \ \forall j,k$. In the setting (9.1), this simply implies to agree on equal $u_k \ \forall k$, while positions λ_{jk} can be arbitrary. This problem is solved by the classical vector space consensus algorithm presented in Section 3.1,

$$\frac{d}{dt}\xi_k^l = \sum_{j \rightsquigarrow k} (\xi_j^l - \xi_k^l) \qquad , \quad k = 1, 2, ..., N \ , \tag{9.2}$$

which exponentially achieves $\xi_k^l = \xi_j^l \ \forall j,k$ if $\mathbb{G}$ is piecewise continuous and uniformly connected. Using (9.1), algorithm (9.2) becomes $\frac{d}{dt}u_k = \sum_{j \rightsquigarrow k} (u_j - u_k)$. To implement its algorithm, agent k relies on the left-invariant velocity ξ_j^l of $j \rightsquigarrow k$ (or alternatively on u_j, which is just the same). The initial values of the u_k can be chosen arbitrarily.

For a time-invariant and undirected communication graph $\mathbb{G}$, (9.2) is a gradient descent for the disagreement cost function $V_r = \sum_k \sum_{j \rightsquigarrow k} \|\xi_k^l - \xi_j^l\|^2$, with the Euclidean metric in $\mathfrak{g}$ (see Section 3.1).

9.1.2 Left-invariant coordination in the fully actuated setting

Left-invariant coordination requires $\xi_k^r = \xi_j^r \ \forall j,k$, which suggests to use

$$\frac{d}{dt}\xi_k^r = \sum_{j \rightsquigarrow k} (\xi_j^r - \xi_k^r) \qquad , \quad k = 1, 2, ..., N \ . \tag{9.3}$$

Algorithm (9.3) can be rewritten in terms of left-invariant variables using (8.1) and Property 2.4.8 (c), eq. (2.6) to get

$$\frac{d}{dt}\xi_k^l = \sum_{j \rightsquigarrow k} (Ad_{g_k^{-1} g_j} \, \xi_j^l - \xi_k^l) \qquad , \quad k = 1, 2, ..., N \ . \tag{9.4}$$

To implement (9.4), agent k must know the relative position $g_k^{-1} g_j$ and velocity ξ_j^l of its in-neighbors $j \rightsquigarrow k$. The initial u_k can be chosen arbitrarily. Since (9.4) is just a change of variables in (9.3), it ensures global exponential synchronization of the ξ_k^r — hence global exponential convergence to LIC — if $\mathbb{G}$ is piecewise continuous and uniformly connected.

The disagreement cost function $V_l = \sum_k \sum_{j \rightsquigarrow k} \|Ad_{g_k} \xi_k^l - Ad_{g_j} \xi_j^l\|^2$ associated to (9.3) is not left-invariant in general (it involves absolute positions g_k), so (9.4) cannot be, in general, a left-invariant gradient of V_l.

Let $\mathcal{G}_u$ be the subclass of compact groups with unitary adjoint representation, i.e. satisfying $\|Ad_g \, \xi\| = \|\xi\| \ \forall g \in G$ and $\forall \xi \in \mathfrak{g}$; for instance, $SO(n) \in \mathcal{G}_u$. It is possible to define a bi-invariant (that is, left- and right-invariant) Riemannian metric on G if and only if $G \in \mathcal{G}_u$, see [6]. The Euclidean metric on left-invariant velocities, which is used in the present developments in accordance with the left-invariant setting, is a left-invariant metric. If $G \in \mathcal{G}_u$, then this metric is bi-invariant, $V_l = \sum_k \sum_{j \rightsquigarrow k} \|Ad_{g_k} \xi_k^l - Ad_{g_j} \xi_j^l\|^2 = \sum_k \sum_{j \rightsquigarrow k} \|\xi_k^l - Ad_{g_k^{-1} g_j} \xi_j^l\|^2$ is bi-invariant and for fixed undirected $\mathbb{G}$, (9.4) is a gradient descent for V_l.

9.1.3 Underactuated LIC and total coordination

In underactuated settings, for RIC , (9.2) makes ξ_k^l move towards the velocities ξ_j^l of its in-neighbors, and thereby ensures that all ξ_k^l stay in the affine space $\mathcal{C}$ if they initially belong to $\mathcal{C}$. But for LIC , (9.4) involves a linear transformation of the ξ_j^l by $Ad_{g_k^{-1}g_j}$ and, a priori, it is possible that $(Ad_{g_k^{-1}g_j}\,\xi_j^l) \notin \mathcal{C}$ for some $j \rightsquigarrow k$; then (9.4) would require ξ_k^l to leave $\mathcal{C}$. Thus algorithm (9.4) cannot always be implemented in an underactuated setting. At equilibrium, (9.4) requires

$$Ad_{\lambda_{jk}}(a + Bu_j) \;=\; a + Bu_k \quad \forall j, k \;, \tag{9.5}$$

for which an underactuated setting not only restricts the current control inputs u_k, but also the relative positions of the agents. Thus a consensus algorithm on velocities is not sufficient, the positions must also be controlled to reach particular λ_{jk}. This issue motivates further study of underactuated LIC in Section 9.3.

There exist underactuated settings for which (9.4) does work to reach LIC . Define $O_{\mathcal{C}} = \{Ad_g\xi : g \in G \text{ and } \xi \in \mathcal{C}\}$. If the Lie group and control setting are such that $O_{\mathcal{C}} \equiv \mathcal{C}$, then LIC imposes no restrictions on agent positions and (9.4) is ensured to work appropriately.

Total coordination requires simultaneous consensus on left- and right-invariant velocities. At equilibrium, this means that (9.5) must hold with equal controls u_k, i.e.

$$Ad_{\lambda_{jk}}(a + Bu_k) \;=\; a + Bu_k \quad \forall j, k \;. \tag{9.6}$$

Even in a fully actuated setting, this puts constraints on the relative positions of the agents. Thus, as for uneractuated LIC , reaching TC requires to coordinate velocities *and* positions. For this reason, total coordination is further studied in Section 9.2.

In the following, it is assumed that the agents are controllable. Obviously, controllability is sufficient for coordination since it allows the agents to reach any position from any initial condition. However, it is not always necessary, as long as positions compatible with (9.5) or (9.6) are globally reachable. In particular, for Abelian groups Ad_g = Identity $\forall g$ so any positions satisfy (9.5) and (9.6); in that case, (underactuated) LIC and TC are trivially solved by the RIC algorithm (9.2) regardless of positions and controllability.

9.2 Control algorithms for fully actuated total coordination

9.2.1 Total coordination on general Lie groups

Total coordination requires to satisfy two objectives, LIC and RIC , simultaneously. In a first step, assume that the agents have at their disposal a reference right-invariant velocity ξ^r which they can track, such that LIC is ensured if $\xi_k^l = Ad_{g_k}^{-1}\xi^r \;\forall k$. It remains to simultaneously achieve RIC , i.e. $\xi_k^l = \xi_j^l \;\forall j, k$, which requires to control relative positions. Writing

$$\xi_k^l = \eta_k^l + \; q_k \;, \quad k = 1, 2, ..., N \;, \tag{9.7}$$

where $\eta_k^l := Ad_{g_k}^{-1}\xi^r$, the question is how to design q_k in order to achieve TC . For fixed undirected communication graph $\mathbb{G}$, inspired by the cost function for RIC , define

$$V_{tr}(g_1, g_2, ..., g_N) = \tfrac{1}{2}\sum_k \sum_{j \rightsquigarrow k} \|\eta_k^l - \eta_j^l\|^2 \;.$$

V_{tr} characterizes the distance from RIC *assuming that every agent implements* $\xi_k^l = Ad_{g_k}^{-1}\xi^r$. The variation of V_{tr} in time due to motion of the g_k is, using Property 2.4.8 (c) eq. (2.7),

$$\frac{d}{dt}V_{tr} = 2\sum_k \sum_{j\rightsquigarrow k} (\eta_k^l - \eta_j^l)\cdot[\eta_k^l,\, \xi_k^l] \tag{9.8}$$

where $\cdot$ denotes the canonical scalar product in $\mathfrak{g}$, that is $a\cdot b = a^T b$ for a and b considered vectors of $\mathbb{R}^n$, or $a\cdot b = \mathrm{trace}(a^T b)$ for a and b considered matrices of $\mathbb{R}^{n\times n}$. Thus if $q_k = 0$ then $\frac{d}{dt}V_{tr} = 0$; a proper choice of $q_k \neq 0$ should allow to decrease V_{tr}. Define the bilinear operator[2] $\langle\,,\,\rangle$ such that $\xi_1\cdot\langle\xi_2,\xi_3\rangle + [\xi_1,\xi_2]\cdot\xi_3 = 0 \quad \forall \xi_1,\xi_2,\xi_3 \in \mathfrak{g}$. Then (9.8) rewrites $\frac{d}{dt}V_{tr} = 2\sum_k\sum_{j\rightsquigarrow k}\langle\eta_k^l, \eta_k^l - \eta_j^l\rangle\cdot q_k$ and the choice

$$q_k = -\langle\eta_k^l,\, \textstyle\sum_{j\rightsquigarrow k}(\eta_k^l - \eta_j^l)\rangle \quad , \quad k = 1,2,...,N\,, \tag{9.9}$$

ensures that V_{tr} is non-increasing along the solutions:

$$\frac{d}{dt}V_{tr} = -2\sum_k\sum_{j\rightsquigarrow k}\langle\eta_k^l,\, \textstyle\sum_{j\rightsquigarrow k}(\eta_k^l - \eta_j^l)\rangle^2 \;\leq 0\,.$$

To obtain an autonomous algorithm for total coordination, it remains to replace the reference velocity ξ^r by agent-related estimates η_k^r on which the agents progressively agree. Since the goal is to agree on a common right-invariant velocity in $\mathfrak{g}$, it is natural to proceed as in Section 9.1.2 and use the consensus algorithm

$$\frac{d}{dt}\eta_k^r = \sum_{j\rightsquigarrow k}(\eta_j^r - \eta_k^r) \tag{9.10}$$

which in terms of left-invariant velocities rewrites, using again Property 2.4.8 (c) eq. (2.7),

$$\frac{d}{dt}\eta_k^l = \sum_{j\rightsquigarrow k}(Ad_{\lambda_{jk}}\,\eta_j^l - \eta_k^l) \;- [\xi_k^l,\, \eta_k^l] \quad , \quad k = 1,2,...,N\,. \tag{9.11}$$

Thus the overall controller is, as depicted on Figure 9.1, the cascade of a consensus algorithm to agree on a desired velocity for LIC , and a position controller designed to decrease a natural distance to RIC . To implement the controller, agent k must receive from in-neighbors $j \rightsquigarrow k$ their relative positions λ_{jk} and the values of their left-invariant auxiliary variables η_j^l.

Figure 9.1: Total coordination as consensus on right-invariant velocity and Lyapunov-based control to right-invariant coordination.

Proposition 9.2.1: *Consider N fully actuated agents communicating along the edges of a connected, fixed, undirected graph $\mathbb{G}$ and evolving on Lie group G according to $\frac{d}{dt}g_k = L_{g_k *}\xi_k^l$ with controller (9.7),(9.9),(9.11).*

[2]In fact, $\langle\,,\,\rangle$ expresses the effect of the Lie bracket on the dual space of $\mathfrak{g}$, and is directly related to the *coadjoint* representation of G; however in general, $\langle\,,\,\rangle$ does not satisfy the Lie bracket properties.

(a) *For any initial conditions* $\eta_k^l(0)$*, the* $\eta_k^r(t)$ *exponentially converge to* $\overline{\eta^r} := \frac{1}{N}\sum_k \eta_k^r(0)$.
(b) *Define* $\overline{V_{tr}}(g_1, g_2, ...g_N) = \frac{1}{2}\sum_k \sum_{j\rightsquigarrow k} \|(Ad_{g_k}^{-1} - Ad_{g_j}^{-1})\overline{\eta^r}\|^2$*. All solutions converge to the critical set of* $\overline{V_{tr}}$*. In particular, left-invariant coordination is asymptotically achieved.*
(c) *Total coordination is (at least locally) asymptotically stable.*

Proof: Since (9.11) is just (9.10) rewritten, (a) is a direct consequence of the convergence properties of vector space consensus algorithms, Propositions 3.1.1 and 3.1.2.

Since the η_k^r converge, (9.7),(9.9) is an asymptotically autonomous system; the autonomous limit system is obtained by replacing $\eta_k^l = Ad_{g_k}^{-1}\overline{\eta^r}$. From the derivation of q_k in (9.9), the limit system is a gradient descent system for $\overline{V_{tr}}(g_1, g_2, ...g_N)$. The latter is smooth because the adjoint representation is smooth. Therefore according to Propositions 2.3.6 and 2.3.8, the limit set of the original system is equal to the set of critical points of $\overline{V_{tr}}$; then $q_k = 0$ and $\xi_k^r = \xi_j^r = \overline{\eta^r}\ \forall j, k$. This proves (b). Then total coordination $\xi_k^l = \xi_j^l\ \forall j, k \Leftrightarrow \overline{V_{tr}} = 0$ is locally asymptotically stable because it is a local (and global) minimum of $\overline{V_{tr}}$. □

An extension to varying and directed $\mathbb{G}$ can be made with additional auxiliary variables, like for consensus on manifolds in Sections 4.3 and 6.2.1. This has indeed been done for $SE(2)$ and $SE(3)$ in [132] and [126]. The procedure depends too much on the particular group to propose general algorithms at this place; the idea is the following. As a first step, consensus algorithms with several auxiliary variables define a desired ξ^r *and* a desired ξ^l. To ensure compatibility of ξ^r and ξ^l, they must both lie on the same adjoint orbit, which is the tricky part; sometimes intuition is needed to express everything in a left-invariant setting. In a second step, cost functions for individual agents are used to ensure that they asymptotically implement the desired velocities. Such algorithms mostly overcome the problem of local minima different from TC , which makes them useful for fixed undirected $\mathbb{G}$ as well.

9.2.2 Total coordination on Lie groups with a bi-invariant metric

When $G \in \mathcal{G}_u$, i.e. G has a bi-invariant metric, then left-invariant controls for LIC can be derived from the cost function $V_l = \sum_k \sum_{j\rightsquigarrow k} \|Ad_{g_k}\xi_k^l - Ad_{g_j}\xi_j^l\|^2$.

A natural idea in this context is to combine the cost functions for LIC and RIC , writing $V_t = V_l + V_r$, and derive a gradient descent for V_t. However, simulations of the resulting control law for $SO(n)$ seem to always converge to $\xi_k^l = 0\ \forall k \in \mathcal{V}$. A possible explanation for this behavior is that this strategy focuses on *velocities*, such that *positions* of the agents are not explicitly controlled, while it is extensively discussed in Section 8.2 how TC at nonzero velocity involves restrictions on compatible positions.

The existence of a bi-invariant metric offers the possibility to switch the roles of LIC and RIC in the method of Section 9.2.1, using a consensus algorithm to define a common *left-invariant* velocity for RIC , and a cost function to drive positions to LIC , as depicted on Figure 9.2.

The consensus algorithm asymptotically defines a common velocity ξ^l by

$$\frac{d}{dt}\eta_k^l = \sum_{j\rightsquigarrow k} (\eta_j^l - \eta_k^l) \quad , \quad k = 1, 2, ..., N\,. \tag{9.12}$$

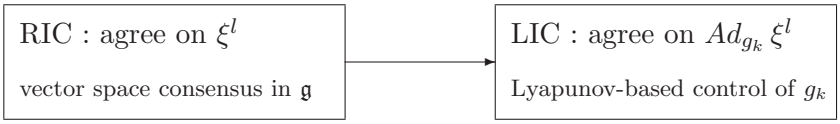


Figure 9.2: Total coordination as consensus on left-invariant velocity and Lyapunov-based control to left-invariant coordination.

Then defining the cost function

$$V_{tl}(g_1, g_2, ..., g_N) = \tfrac{1}{2}\sum_k \sum_{j\rightsquigarrow k} \|Ad_{g_k}\,\eta_k^l - Ad_{g_j}\,\eta_j^l\|^2 = \tfrac{1}{2}\sum_k \sum_{j\rightsquigarrow k} \|\eta_k^l - Ad_{g_k^{-1}g_j}\,\eta_j^l\|^2$$

for LIC and proceeding as in Section 9.2.1, one obtains controller (9.7) with (mind the sign)

$$q_k = \langle \eta_k^l,\ \textstyle\sum_{j\rightsquigarrow k}\,(\eta_k^l - Ad_{g_k^{-1}g_j}\,\eta_j^l)\rangle\ . \tag{9.13}$$

Proposition 9.2.2: *Consider N fully actuated agents communicating along the edges of a connected, fixed, undirected graph $\mathbb{G}$ and evolving on Lie group $G \in \mathcal{G}_u$ according to $\frac{d}{dt}g_k = L_{g_k *}\xi_k^l$ with controller (9.7),(9.12),(9.13).*

(a) *For any initial conditions $\eta_k^l(0)$, the $\eta_k^l(t)$ exponentially converge to $\overline{\eta^l} := \frac{1}{N}\sum_k \eta_k^l(0)$.*

(b) *Define $\overline{V_{tl}}(g_1, g_2, ...g_N) := \frac{1}{2}\sum_k \sum_{j\rightsquigarrow k} \|(Ad_{g_k} - Ad_{g_j})\overline{\eta^l}\|^2$. All solutions converge to the critical set of $\overline{V_{tl}}$. In particular, right-invariant coordination is asymptotically achieved.*

(c) *Total coordination is (at least locally) asymptotically stable.*

Proof: The proof is omitted because it is similar to the proof of Proposition 9.2.1. □

An advantage of the alternative controller (9.12),(9.13) over controller (9.9),(9.11) of Section 9.2.1 is that the control design can be directly extended to underactuated agents. Indeed, (9.12) defines a valid consensus velocity $\xi^l \in \mathcal{C} = \{a + Bu : u \in \mathbb{R}^m\}$ for underactuated agents provided that $\eta_k^l(0) \in \mathcal{C}\ \ \forall k \in \mathcal{V}$. The only change to be made is that q_k, instead of the exact gradient descent in (9.13), is its projection onto the control range of B:

$$\xi_k^l = a + Bu_k = \eta_k^l + B\,B^T q_k \tag{9.14}$$

with q_k defined by (9.13), assuming without loss of generality that the columns of B are orthonormal vectors. In order to extend Proposition 9.2.2 to underactuated agents, the convergence argument for asymptotically autonomous systems must be extended to projections of the gradient system (9.13); a general proof of this technical issue is presently lacking.

The double-bracket flow has been developed by R. Brockett, see [19] for instance, as a general form for gradient algorithms on adjoint orbits of compact semi-simple groups, using the bi-invariant *Killing metric* (see [19] for a definition).

This is directly connected to the algorithms of the present section, because once the consensus algorithm has converged, the gradient control for agent positions involves a cost function on the adjoint orbit of $\overline{\eta^l}$ or $\overline{\eta^r}$. One example in [19] minimizes the distance towards a subset of $\mathfrak{g}$. A similar objective is pursued in Section 9.3 for underactuated LIC , although with

a different class of subsets. A main difference of [19] is its focus on the evolution of variables in $\mathfrak{g}$, disregarding the underlying group, while in the present dissertation the primary goal is to actually control the positions of (possibly underactuated) agents on G.

If $G \in \mathcal{G}_u$ and the bi-invariant Killing metric coincides with the left-invariant metric of the present setting, then $\langle\,,\,\rangle = -[\,,\,]$ and e.g. control (9.9) for g_k implies that η_k^l follows the double bracket flow

$$\tfrac{d}{dt}\eta_k^l = [\eta_k^l,\, [\eta_k^l,\, \textstyle\sum_{j\rightsquigarrow k}(\eta_k^l - \eta_j^l)]] \,. \tag{9.15}$$

for $\eta_k^r = \xi^r$ fixed. This holds among others for the following example in $SO(3)$.

Ex. 9.2.3: controller for total coordination in $SO(3)$: Control laws for coordination in $SO(3)$ abound in the literature — see among others the papers about satellite attitudes in the Introduction of Parts II and III. Total coordination on $SO(3)$ requires aligned rotation axes, and thus synchronizes attitudes up to a phase around the rotation axis (see Figure 8.1).

$SO(3) \in \mathcal{G}_u$, so the approach of Section 9.2.2 can be applied. Algorithm (9.12) is used verbatim, with $\eta_k^l \in \mathbb{R}^3$ the auxiliary variable associated to angular velocity ω_k^l. Since $\langle\,,\,\rangle = -[\,,\,]$ and $[\,,\,]$ is the vector product in $\mathbb{R}^3$, in the fully actuated case (9.7),(9.13) lead to

$$\omega_k^l = \eta_k^l \; + \eta_k^l \wedge \left(\sum_{j\rightsquigarrow k} Q_k^T Q_j \eta_j^l \right) \quad , \quad k = 1, 2, ..., N \,. \tag{9.16}$$

Proposition 9.2.2 can be strengthened as follows for specific graphs.

Proposition 9.2.4: *Consider the setting of Proposition 9.2.2 on $SO(3)$. If $\mathbb{G}$ is an undirected tree or the complete graph, then total coordination is the only asymptotically stable limit set.*

Proof: It must be shown that TC is the only local minimum of V_{tl}. Fixing $\eta_k^l = \omega^l \; \forall k$, critical points of $V_{tl}(g_1, g_2, ..., g_N)$ correspond to

$$(Q_k\omega^l) \wedge (\textstyle\sum_{j\rightsquigarrow k} Q_j\omega^l) = 0 \quad \forall k \in \mathcal{V} \,. \tag{9.17}$$

For the tree, fix an arbitrary root and start with the leaves c. Then $(Q_c\omega^l)\wedge(Q_p\omega^l) = 0$ where p is the parent of c. As a consequence, (9.17) for the parent becomes $(Q_p\omega^l) \wedge (Q_{pp}\omega^l) = 0$ where pp is the parent of p. Using this argument up to the root, all $(Q_k\omega^l)$ must be parallel and $Q_k\omega^l = \pm\alpha \; \forall k \in \mathcal{V}$, for some $\alpha \in \mathbb{R}^3 : \|\alpha\| = \|\omega^l\|$. Partition the agents in two groups corresponding to $+\alpha$ and $-\alpha$; then rotating those groups towards each other decreases V_{tl}, such that the situation is unstable unless one of the groups is empty, meaning TC is achieved. For the complete graph, (9.17) becomes $(Q_k\omega^l)\wedge\psi = 0 \; \forall k \in \mathcal{V}$, where $\psi = \sum_j Q_j\omega^l$. This implies that either all $Q_k\omega^l$ are parallel or $\psi = 0$. In the first case, further discussion is as for the tree. Rewriting $V_{tl} = N^2 \; \|\omega^l\|^2 - \frac{1}{2}\|\psi\|^2$ shows that $\psi = 0$ corresponds to a maximum of V_{tl}. □

It is straightforward to adapt (9.16) for underactuated agents. A popular underactuation is to consider two orthogonal axes of allowed rotations $\mathbf{e}_1$ and $\mathbf{e}_2$, either controlling both, i.e. $\omega_k^l = u_1\mathbf{e}_1 + u_2\mathbf{e}_2$, or imposing a fixed rotation rate around one, i.e. $\omega_k^l = \mathbf{e}_1 + u_2\mathbf{e}_2$. Both cases are controllable (see [60]), so the Jurdjevic-Quinn theorem (see Proposition 2.3.3) ensures asymptotic stability of TC *if* $\eta_k^l = \overline{\eta^l} \; \forall k$ *after finite time.* As for the general theory (Proposition 9.2.2), a formal convergence proof for the asymptotically autonomous case, where the η_k^l follow (9.12) and only *asymptotically* agree, is currently missing. ◇

O.Q.: A formal convergence proof is currently missing for the adaptation to *underactuated* agents of the TC algorithms on Lie groups proposed in Section 9.2.2. The reason for this is the lack of convergence results for asymptotically autonomous systems whose asymptotic dynamics are the projection of a gradient system onto the range of an underactuated controller.

9.3 Control algorithms for underactuated left-invariant coordination

Total coordination may appear as a rather academic example. However, the methodology developed in Section 9.2 for TC control design is instrumental to achieve left-invariant coordination of underactuated agents, which is an ubiquitous problem in practical applications. Here the role of the cost function is no longer to add a second level of coordination, but to take the underactuation constraints into account. The present section explicitly covers possibly directed and time-varying interconnection graphs.

The controller for underactuated LIC is decomposed in the two steps illustrated on Figure 9.3. A feasible common right-invariant velocity is determined by a consensus algorithm. Then to ensure that each agent can actually apply it, a Lyapunov-based feedback decreases the distance of the corresponding left-invariant velocity to $\mathcal{C} = \{a + Bu : u \in \mathbb{R}^m\}$.

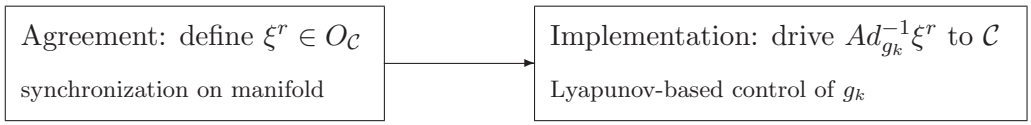


Figure 9.3: Underactuated left-invariant coordination as constrained consensus on desired right-invariant velocity and Lyapunov-based control to left-invariant coordination.

The consensus algorithm must reach agreement on a vector ξ^r in the set

$$O_\mathcal{C} := \{Ad_g\xi : \xi \in \mathcal{C} \text{ and } g \in G\}.$$

If $O_\mathcal{C}$ is convex, then it is sufficient to initialize the consensus algorithm (9.11) with $\eta_k^l(0) \in \mathcal{C}$. When $O_\mathcal{C}$ is not convex, the consensus algorithm must be adapted and the present work has no general method to propose. Strategies inspired from Part II of the present dissertation for compact homogeneous manifolds may be helpful, as illustrated in an example below.

O.Q.: It is currently unclear if an easy general method can be proposed for the design of left-invariant consensus algorithms to properly agree on a desired right-invariant velocity $\xi^r \in O_\mathcal{C}$ for underactuated left-invariant coordination.

Assuming that a feasible right-invariant velocity ξ^r is agreed on, the design of a Lyapunov based control to left-invariant coordination proceeds similarly to Section 9.2.1. Denote by $d(\eta, \mathcal{C}) := \min_{\zeta\in\mathcal{C}} \|\eta - \zeta\|$ the Euclidean distance in $\mathfrak{g}$ from η to the set $\mathcal{C}$. The projection of η on $\mathcal{C}$ is $\Pi_\mathcal{C}(\eta) := \text{argmin}_{\zeta\in\mathcal{C}} \|\eta - \zeta\|$. Since $\mathcal{C}$ is convex, this is a single point. Following the

same steps as in Section 9.2.1, define $\eta_k^l := Ad_{g_k}^{-1}\xi^r$. Writing

$$\xi_k^l = a + Bu_k = \Pi_{\mathcal{C}}(\eta_k^l) + Bq_k \quad , \quad k = 1, 2, ..., N \, , \tag{9.18}$$

the task is to design $q_k \in \mathbb{R}^m$ such that asymptotically, g_k is driven to a point where $\eta_k^l \in \mathcal{C}$ and q_k converges to 0. For each individual agent k, define the cost function

$$V_k(g_k) = \tfrac{1}{2}\|Ad_{g_k}^{-1}\xi^r - \Pi_{\mathcal{C}}(Ad_{g_k}^{-1}\xi^r)\|^2 = \tfrac{1}{2}\|\eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l)\|^2 \, . \tag{9.19}$$

V_k characterizes the distance from η_k^l to $\mathcal{C}$, that is the distance to LIC *assuming that every agent implements* $\xi_k^l = \Pi_{\mathcal{C}}(Ad_{g_k}^{-1}\xi^r)$. The variation of V_k in time due to motion of g_k is

$$\tfrac{d}{dt}V_k = (\eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l))^T \, [\eta_k^l, \, \Pi_{\mathcal{C}}(\eta_k^l) + Bq_k] \tag{9.20}$$

with $\mathfrak{g} \cong \mathbb{R}^m$. Going further along the lines of Section 9.2.1 requires to assume that the control setting (pair a, B) and Lie algebra structure are such that $(\eta - \Pi_{\mathcal{C}}(\eta))^T[\eta, \, \Pi_{\mathcal{C}}(\eta)] \leq 0$ $\forall \eta \in O_{\mathcal{C}}$. Then, assuming without loss of generality that the columns of B are orthonormal vectors, (9.20) implies $\frac{d}{dt}V_k \leq q_k^T \, B^T \langle \eta_k^l, \, \eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l)\rangle$ and a natural control is

$$q_k = -B^T \langle \eta_k^l, \, \eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l)\rangle \quad , \quad k = 1, 2, ..., N \, . \tag{9.21}$$

When $O_{\xi^r} \subseteq \mathcal{C}$, position control is unnecessary and (9.18) reduces to $\xi_k^l = Ad_{g_k}^{-1}\xi^r \ \forall t$.

In the overall controller, agents only interact through the consensus algorithm, not through the cost function. Therefore assumptions on $\mathbb{G}$ depend on the consensus algorithm, which should ideally allow a uniformly connected, directed and time-varying graph. It is anticipated that this will require agent k to get from agents $j \rightsquigarrow k$ their relative positions λ_{jk} and values of auxiliary variables η_j^l (maybe actually several other variables with larger total dimension).

A general characterization of solutions of the closed-loop system is difficult because the position controller is not a gradient. The following result involves mostly technical assumptions that can be readily checked for any particular case.

Proposition 9.3.1: *Consider N underactuated agents communicating along the edges of a uniformly connected graph $\mathbb{G}$ and evolving on Lie group G according to $\frac{d}{dt}g_k = L_{g_k *}\xi_k^l$ with controller (9.18),(9.21). Assume that $(\eta - \Pi_{\mathcal{C}}(\eta))^T[\eta, \, \Pi_{\mathcal{C}}(\eta)] \leq 0 \ \forall \eta \in O_{\mathcal{C}}$. Assume that the η_k^l, $k = 1, 2, ..., N$, are driven by an appropriate consensus algorithm such that, independently of the agent motions $g_k(t)$, the $Ad_{g_k}\eta_k^l$ exponentially agree on $\xi^r \in O_{\mathcal{C}} \ \forall k \in \mathcal{V}$.*

(a) *If the agents are controllable, then LIC is locally asymptotically stable.*

(b) *If, for any fixed ξ^r and $\eta_k^l = Ad_{g_k}^{-1}\xi^r$, $B^T\langle \eta_k^l, \, \eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l)\rangle$ continuously approaching 0 implies g_k continuously approaching $\mathcal{H} = \{g : B^T\langle Ad_g^{-1}\xi^r, \, Ad_g^{-1}\xi^r - \Pi_{\mathcal{C}}(Ad_g^{-1}\xi^r)\rangle = 0\}$, and bounded V_k implies bounded η_k^l, then all agent trajectories on G converge to $\mathcal{H}$.*

Proof: The overall system is a cascade of the exponentially stable consensus algorithm and position controller (9.18),(9.21) which is decoupled for the individual agents. Consider the position controller assuming $\eta_k^r = \xi^r$ fixed. $V_k(g_k)$ is non-increasing in open loop (i.e. for $q_k = 0$), and q_k implements the gradient descent of V_k along the range of B. Therefore, if the agents are controllable, the Jurdjevic-Quinn theorem (see Proposition 2.3.3) implies local asymptotic stability of the minimum $V_k = 0$. Then for η_k^r asymptotically converging to ξ^r, Proposition 2.3.4 can be applied, identifying $y = \eta_k^r - \xi^r$, $x = g_k$, $f(x)$ the position controller

with η_k^r fixed to ξ^r, and $h(x,y)$ the difference due to $\eta_k^r - \xi^r \neq 0$. This directly allows to conclude that $V_k = 0\ \forall k$ is locally asymptotically stable for the overall system, proving (a).

To prove (b), first consider the case with constant $\eta_k^r = \xi^r$. Then V_k can only decrease, and since it is bounded from below it tends to a limit; thus $\frac{d}{dt}V_k$ is Riemann integrable over $t \in (0, +\infty)$. Since V_k is bounded, η_k^l is bounded; then $\frac{d^2}{dt^2}V_k$, which is a continuous function of η_k^l, is also bounded, such that $\frac{d}{dt}V_k$ is uniformly continuous in time for $t \in (0, +\infty)$. Therefore Barbalat's Lemma implies that $\frac{d}{dt}V_k$ converges to 0, implying that $B^T\langle \eta_k^l,\, \eta_k^l - \Pi_{\mathcal{C}}(\eta_k^l)\rangle$ converges to 0, concluding the proof. Now in fact η_k^r is not constant but only exponentially converges to $\xi^r\ \forall k \in \mathcal{V}$. But this changes nothing to the fact that V_k tends to a finite limit and $\frac{d^2}{dt^2}V_k$ is bounded, so the same argument applies. □

Condition $(\eta - \Pi_{\mathcal{C}}(\eta))^T[\eta,\, \Pi_{\mathcal{C}}(\eta)] \leq 0$ in Proposition 9.3.1 does not always hold for systems with a nonzero drift $a \neq 0$, see academic Example 9.3.2 below. However, it is often satisfied in practice, like for steering control of rigid bodies in the next Example 9.3.3.

Ex. 9.3.2: a setting violating the assumptions of Proposition 9.3.1: Consider a set of agents evolving on $SE(3)$ with the control setting

$$(v_k^l,\, \omega_k^l) = a + Bu_k = (\mathbf{e_1},\, 0) + (0,\, u_{k\,1}\mathbf{e_1} + u_{k\,2}\mathbf{e_2}) \tag{9.22}$$

where $\mathbf{e_1}$ and $\mathbf{e_2}$ are the first two vectors of an orthonormal basis in $\mathbb{R}^3$. The whole Lie algebra can be rebuilt from the vectors of $\mathcal{C} = \{(\mathbf{e_1},\, \lambda_1\mathbf{e_1} + \lambda_2\mathbf{e_2}) : \lambda_1 \in \mathbb{R} \text{ and } \lambda_2 \in \mathbb{R}\}$ by combining additions and Lie brackets. Therefore, see [60] for instance, the agents are controllable. With $\eta = (v_1\mathbf{e_1} + v_2\mathbf{e_2} + v_3\mathbf{e_3},\, w_1\mathbf{e_1} + w_2\mathbf{e_2} + w_3\mathbf{e_3}) \in \mathfrak{g}$, $(\eta - \Pi_{\mathcal{C}}(\eta))^T\,[\eta,\, \Pi_{\mathcal{C}}(\eta)]$

$$\begin{aligned}
&= (w_3\mathbf{e_3})^T\ ((w_1\mathbf{e_1} + w_2\mathbf{e_2} + w_3\mathbf{e_3}) \wedge w_3\mathbf{e_3}) \\
&\quad + ((v_1 - 1)\mathbf{e_1} + v_2\mathbf{e_2} + v_3\mathbf{e_3})^T\ ((w_1\mathbf{e_1} + w_2\mathbf{e_2}) \wedge (v_1\mathbf{e_1} + v_2\mathbf{e_2} + v_3\mathbf{e_3})) \\
&\quad - ((v_1 - 1)\mathbf{e_1} + v_2\mathbf{e_2} + v_3\mathbf{e_3})^T\ ((w_1\mathbf{e_1} + w_2\mathbf{e_2} + w_3\mathbf{e_3}) \wedge (\mathbf{e_1})) \\
&= -(w_2v_3 + w_3v_2)\,.
\end{aligned}$$

Now choose $u_{k\,1} = 0$, $u_{k\,2} = \alpha \neq 0$ and $g = (I_3,\, r_1\mathbf{e_1})$ to define a specific $\eta = Ad_g(a + Bu_k) = (\mathbf{e_1} + r_1u_{k\,2}\mathbf{e_3},\, u_{k\,2}\mathbf{e_2}) \in O_{\mathcal{C}}$. Then $(\eta - \Pi_{\mathcal{C}}(\eta))^T\,[\eta,\, \Pi_{\mathcal{C}}(\eta)] = -r_1(u_{k\,2})^2$ can be positive for some r_1. ◇

Ex. 9.3.3: steering control on $SE(3)$: Left-invariant coordination on $SE(3)$ under steering control is studied in [123, 126]. The present section shows how the algorithms of [126] follow from the present general framework.

As shown in Example 8.2.9, "moving in formation", i.e. such that the relative position and orientation of agent j with respect to agent k are fixed in the reference frame of agent k, $\forall j, k \in \mathcal{V}$, is equivalent to requiring LIC . Linear and angular velocity in body frame correspond to the components (v_k^l, ω_k^l) of ξ_k^l. So the problem of controlling each agent in its own frame, with feedback involving *relative* positions and orientations of other agents only, fits the left-invariant problem setting. The constraint of *steering control* — i.e. fixed linear velocity in agent frame — implies (9.1) of the form $\xi_k^l = (v_k^l,\, \omega_k^l) = a + Bu_k = (\mathbf{e}_1, u_k)$ and thus $\mathcal{C} = (\mathbf{e}_1, \mathbb{R}^3)$. Steering controlled agents on $SE(3)$ are controllable, see [60].

Following the method of Section 9.3, write auxiliary variables $\eta_k^l = (\eta_{v\,k}^l, \eta_{\omega\,k}^l)$. Then $\Pi_{\mathcal{C}}(\eta_k^l) = (\mathbf{e}_1, \eta_{\omega\,k}^l)$, cost function $V_k = \frac{1}{2}\|\eta_{v\,k}^l - \mathbf{e}_1\|^2$ and straightforward calculations show

that (9.20) becomes $\frac{d}{dt}V_k = q_k^T(\eta_{v\,k}^l \wedge \mathbf{e}_1)$. This means that $(\eta - \Pi_{\mathcal{C}}(\eta))^T[\eta, \Pi_{\mathcal{C}}(\eta)] = 0$ $\forall \eta \in O_{\mathcal{C}}$, such that the condition of Proposition 9.3.1 is satisfied. Then (9.18),(9.21) yield

$$u_k = \eta_{\omega\,k}^l + \mathbf{e}_1 \wedge \eta_{v\,k}^l \quad , \quad k = 1, 2, ..., N\,. \tag{9.23}$$

This is the same control law as derived in [126] from intuitive arguments. If an appropriate consensus algorithm is built, then Proposition 9.3.1 implies local asymptotic stability of 3-dimensional "motion in formation" with steering control (9.23); in fact, [126] improves Proposition 9.3.1 by also showing that *globally*, LIC is the only stable limit set.

It remains to design a consensus algorithm for the η_k^l. The solution of [126] is just included here for completeness. Like in Example 8.2.9, two cases are distinguished: linear motion $\omega^r = 0$ and helicoidal (of which a special case is circular) motion $\omega^r \neq 0$. The first case is non-generic for arbitrary $\eta_k^l(0)$, but if coordinated motion on parallel straight lines is desired it can be ensured by imposing $\eta_{\omega\,k}^l(0) = 0\ \forall k$.

- If $\eta_{\omega\,k}^l = 0$ (linear motion), then $\eta_{v\,k}^l = Q_k^T \eta_{v\,k}^r$ and $O_{(\mathbf{e}_1,0)} = \{(\lambda, 0) \in \mathbb{R}^3 \times \mathbb{R}^3 : \|\lambda\| = 1\}$. Agreement on v^r in the unit sphere can be achieved following the method of Part II, Section 6.2.1, by normalizing the result of a consensus algorithm in $\mathbb{R}^3$; in fact normalizing is not even necessary, since it would just change the gain in (9.23). This leads to
$$\frac{d}{dt}\eta_{v\,k}^l = \sum_{j \rightsquigarrow k}(Q_k^T Q_j \eta_{v\,j}^l - \eta_{v\,k}^l) - u_k \wedge \eta_{v\,k}^l \quad , \quad k = 1, 2, ..., N\,. \tag{9.24}$$
- If $\eta_{\omega\,k}^l \neq 0$ (helicoidal motion), then $\eta_{\omega\,k}^l = Q_k^T \eta_{\omega\,k}^r$ and $\eta_{v\,k}^l = Q_k^T \eta_{v\,k}^r - (Q_k^T r_k) \wedge (Q_k^T \eta_{\omega\,k}^r)$, and $O_{\mathcal{C}} = \{(\gamma + \beta \wedge \alpha,\, \alpha) : \alpha, \beta, \gamma \in \mathbb{R}^3 \text{ and } \|\gamma\| \leq 1\}$. It is not easy to design a consensus algorithm that achieves agreement on $\xi^r \in O_{\mathcal{C}}$ *and* that can be written with left-invariant variables. Again the solution is to enlarge the overall dimension of the variables used for the consensus algorithm with respect to the dimension of $O_{\mathcal{C}}$. The algorithm proposed in [126] replaces η_k^l by three components $\alpha_k = \eta_{\omega\,k}^l \in \mathbb{R}^3$, $\beta_k \in \mathbb{R}^3$ and $\gamma_k \in \mathbb{R}^3$ such that $\eta_k^l = (\gamma_k + \beta_k \wedge \alpha_k,\, \alpha_k)$. Left-invariant consensus algorithms can then be decoupled for the α_k, the β_k and the γ_k. With the notations of the present dissertation, the corresponding consensus algorithm proposed in [126] is
$$\begin{aligned}
\tfrac{d}{dt}\alpha_k &= \sum_{j \rightsquigarrow k}(Q_k^T Q_j \alpha_j - \alpha_k) \; - u_k \wedge \alpha_k \\
\tfrac{d}{dt}\beta_k &= \sum_{j \rightsquigarrow k}(Q_k^T Q_j \beta_j - \beta_k + Q_k^T(r_j - r_k)) \; - u_k \wedge \beta_k - \mathbf{e}_1 \\
\tfrac{d}{dt}\gamma_k &= \sum_{j \rightsquigarrow k}(Q_k^T Q_j \gamma_j - \gamma_k) \; - u_k \wedge \gamma_k \qquad\qquad , \quad k = 1, 2, ..., N\,.
\end{aligned}$$
Comparing the terms and factors appearing in this algorithm with left-invariant relative position $g_k^{-1} g_j = (Q_k^T(r_j - r_k),\, Q_k^T Q_j)$, one observes that the algorithm is indeed left-invariant. Computing the associated evolution of $\eta_k^r = Ad_{g_k}(\gamma_k + \beta_k \wedge \alpha_k,\, \alpha_k)$, one verifies that it is indeed an exponential consensus algorithm for the η_k^r (see [126]).

$\diamond$

Ex. 9.3.4: analogy with $SO(3)$: Linear motion under steering control on $SE(3)$ requires to align vectors $Q_k \mathbf{e}_1$ for all agents. This is in fact equivalent to total coordination on $SO(3)$ with $\eta_k^l = \omega^l = \mathbf{e}_1\ \forall k \in \mathcal{V}$. Example 9.3.3 thus illustrates a method for fully actuated TC on $SO(3)$ with *uniformly connected* $\mathbb{G}$ (instead of fixed undirected $\mathbb{G}$ as in Section 9.2). $\diamond$

Ex. 9.3.5: steering control in $SE(2)$; linking LIC with TC: LIC under steering control on $SE(2)$ is treated in [131, 132]. The algorithms obtained intuitively, and recovered with the present general method, are a simplification to lower dimension of the algorithms for $SE(3)$.

In fact, the setting of steering control on $SE(2)$ is such that $\forall g \in SE(2)$ and $\forall\, u \in \mathbb{R}$,

$$Ad_g(a + Bu) = f(g, u) + Bu \quad \text{with} \quad f(g, u) \text{ orthogonal to } Bu\ . \tag{9.25}$$

Explicitly, $a + Bu = (\mathbf{e}_1,\, 0) + (0,\, u) \in \mathbb{R}^2 \times \mathbb{R}$ and $Ad_g(\mathbf{e}_1, u) = (Q_\theta \mathbf{e}_1 - uQ_{\pi/2} r,\, 0) + (0,\, u)$. Then LIC automatically implies equal u_k, thus RIC , meaning that *underactuated LIC is equivalent to TC*. This holds for any group and control setting satisfying (9.25).

For steering control on $SE(3)$, LIC is slightly different from TC because $Ad_g(\mathbf{e}_1, u) = (Q\mathbf{e}_1 + r \wedge (Qu), Qu)$, so (9.25) would require $(Qu)^T(Q\mathbf{e}_1) = u^T\mathbf{e}_1 = 0$ which is not true in general. For LIC under steering control on $SE(3)$, the $\omega_k^l = u_k$ can differ by arbitrary rotations around $\mathbf{e}_1$, while TC requires equal ω_k^l. ⋄

9.4 Coordinated motion with particular configurations

In the developments presented so far, the objective is to reach a motion where relative positions are fixed, without really caring about the maintained *configuration*, i.e. the actual values of the relative positions between agents; the only requirement is that they must be compatible with TC or underactuated LIC . However, applications often require to stabilize particular configurations which are more efficient than others e.g. for sensing or power consumption (see Section 1.1 in the Introduction); at least collisions have to be avoided. The present section examines how to combine the framework for coordinated motion with the framework for stabilizing specific configurations as developed in Part II of this dissertation.

Motivated by examples 8.2.8, 8.2.9 and 8.2.10, "motion in formation" is interpreted as left-invariant coordination, which means that a configuration is defined as a specific set of *left-invariant* relative positions. The developments of Part II for defining and reaching particular configurations are valid for connected *compact* Lie groups or homogeneous manifolds. The following considers fixed undirected graphs in order to reach consensus, anti-consensus or balanced configurations with the gradient method of Section 6.1; it is as always possible to use additional auxiliary variables in order to extend the method to uniformly connected graphs. Although the developments are made for left-invariant coordination, total coordination can be discussed along similar lines.

9.4.1 Fully actuated agents

Fully actuated agents can follow a left-invariant coordinated motion with any configuration. Therefore a priori, the g_k can be distributed arbitrarily on the Lie group G and in order to apply the developments of Part II, G must be compact. For notational convenience, the following considers the case of $SO(n)$. This allows to address for instance motion on the sphere as in Example 8.2.10. The basic idea is, as in the previous sections, to combine a consensus algorithm defining a desired velocity with a cost function to control the agents' positions, see Figure 9.4.

Agreement on a desired right-invariant velocity ξ^r is achieved as previously with

$$\tfrac{d}{dt}\eta_k^l = \sum_{j \rightsquigarrow k} (Q_k^T Q_j\, \eta_j^l\, Q_j^T Q_k\ - \eta_k^l)\ - \xi_k^l\, \eta_k^l + \eta_k^l\, \xi_k^l \quad , \quad k = 1, 2, ..., N\,, \tag{9.26}$$

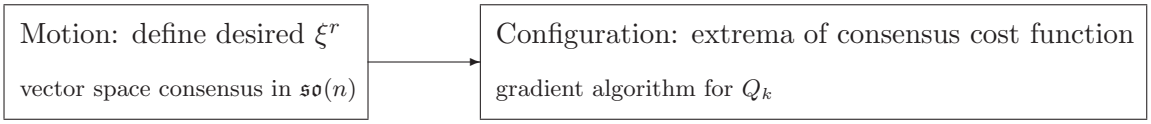


Figure 9.4: Achieving coordinated motion with consensus configurations for fully actuated agents on $SO(n)$, thanks to consensus defining the motion and an optimization setting of Sections 5.3 and 6.1 defining the configuration.

where $\eta_k^l \in \mathfrak{so}(n)$ are auxiliary variables and $\xi_k^l \in \mathfrak{so}(n)$ are control inputs (= actual velocities), both belonging to the space of $n \times n$ antisymmetric matrices. Then ξ_k^l is subdivided into

$$\xi_k^l = \eta_k^l + q_k \quad , \quad k = 1, 2, ..., N \,, \tag{9.27}$$

where q_k is designed to asymptotically stabilize a set of appropriate configurations.

In Section 5.3, it is shown that local maxima of the cost function

$$V_L = \tfrac{1}{2} \sum_{k=1}^{N} \sum_{j \rightsquigarrow k} a_{jk} \, \mathrm{trace}(Q_j^T Q_k) \,, \tag{9.28}$$

correspond to *consensus configurations* on $SO(n)$. Local minima are *anti-consensus configurations*, and *balanced configurations* seem to be directly linked to the minima of (9.28) for the complete graph. The only maximum of (9.28) for the complete graph is synchronization, i.e. all agents at the same position $Q_1 = Q_2 = ... = Q_N$. In Section 6.1, gradient algorithms are designed to locally asymptotically stabilize these configurations; this amounts to using (9.27) with $\eta_k^l = 0 \; \forall k \in \mathcal{V}$ and

$$q_k = \alpha \sum_{j \rightsquigarrow k} \left(Q_k^T Q_j - Q_j^T Q_k \right) \quad , \quad k = 1, 2, ..., N \,. \tag{9.29}$$

with $\alpha > 0$ (respectively $\alpha < 0$) for consensus (respectively anti-consensus) configurations. The following shows that this also works when $\eta_k^l \neq 0$.

Proposition 9.4.1: *Consider a swarm of N fully actuated agents communicating along the edges of a connected, fixed undirected graph $\mathbb{G}$ and evolving on Lie group $SO(n)$ according to $\frac{d}{dt} Q_k = Q_k \xi_k^l$ with controller (9.26),(9.27),(9.29). Then for all initial conditions, the swarm converges to LIC with a configuration in the critical set of V_L. For $\alpha > 0$ (resp. $\alpha < 0$), all solutions in the* asymptotically stable *limit set consist of LIC with a consensus (resp. anti-consensus) configuration for graph $\mathbb{G}$.*

Proof: Algorithm (9.26) is a consensus algorithm for the $\eta_k^r := Ad_{Q_k} \eta_k^l = Q_k \eta_k^l Q_k^T$ in the vector space $\mathfrak{so}(n)$, independent of the agents' motion. Thus Proposition 3.1.1 ensures convergence of the η_k^r towards a common value $\eta_1^r = \eta_2^r = ... = \eta_N^r =: \xi^r$ for all initial conditions. Then (9.27), (9.29) with $\frac{d}{dt} Q_k = Q_k \xi_k^l$ is an asymptotically autonomous system, where the limiting autonomous system is obtained by replacing η_k^l with $Ad_{Q_k^{-1}} \xi^r = Q_k^T \xi^r Q_k$.

The evolution of V_L when agreement on a desired right-invariant velocitiy is achieved, i.e. under $\xi_k^l = Q_k^T \xi^r Q_k + q_k$, is strictly equivalent to its evolution under $\xi_k^l = q_k$, for any

ξ^r. Indeed, Proposition 8.2.1 ensures that, quasi by definition, left-invariant relative positions $Q_k^T Q_j$ do not change when the same signal is added to the right-invariant velocity of all agents. But when $\xi_k^l = q_k$, the limiting autonomous system is the gradient algorithm derived in Section 6.1 for V_L. Therefore, Propositions 2.3.6 and 2.3.8 ensure that all solutions of the complete system converge to the set of critical configurations of V_L; then $q_k = 0$ and LIC is achieved since asymptotically $\xi_k^r = \eta_k^r = \xi^r \ \forall k \in \mathcal{V}$. For $\alpha > 0$ (resp. $\alpha < 0$), the stable configurations are the maxima (resp. minima) of V_L, which with Proposition 5.3.2 concludes the proof. □

9.4.2 Underactuated agents

Inspired by the two-step strategy used in previous sections, consider that agreement on a common desired right-invariant velocity $\xi^r \in O_{\mathcal{C}} = \{Ad_g \xi : g \in G \text{ and } \xi \in \mathcal{C}\}$ has been achieved with a proper consensus algorithm. Before considering the achievement of particular configurations, the set $\mathcal{K} := \{g \in G : Ad_g^{-1} \xi^r \in \mathcal{C}\}$ of positions that are *compatible* with ξ^r must be investigated. By definition, velocities of the form $Ad_g^{-1}\xi^r$ with $g \in \mathcal{K}$ belong to $O_{\xi^r} \cap \mathcal{C}$. For each $\xi^l \in O_{\xi^r} \cap \mathcal{C}$, the set of positions g for which $Ad_g^{-1}\xi^r = \xi^l$ can be written as a right translation of the isotropy group $CM_{\xi^r} = \{g \in G : Ad_g \xi^r = \xi^r\}$ by some specific element g^* satisfying $Ad_{g^*}^{-1}\xi^r = \xi^l$. Indeed,

- take one position g^* for which $Ad_{g^*}^{-1}\xi^r = \xi^l$;
- by definition of a group, all elements $g \in G$ — a fortiori those for which $Ad_g^{-1}\xi^r = \xi^l$ — can be written as $g = hg^*$ for some appropriate $h \in G$;
- this leads to

$$Ad_{g^*}^{-1} Ad_h^{-1} \xi^r = Ad_{g^*}^{-1}\xi^r \quad \Leftrightarrow \quad Ad_h^{-1}\xi^r = \xi^r$$

by invertibility of $Ad_{g^*}^{-1}$, such that h must belong to the isotropy group CM_{ξ^r}.

Thus $\mathcal{K}$ is the union over all $\xi^l \in O_{\xi^r} \cap \mathcal{C}$ of right translated versions $CM_{\xi^r} g^*$ of isotropy group CM_{ξ^r}, where g^* depends on ξ^l. The adjoint orbit O_{ξ^r} is a homogeneous manifold, but at the present point it is unclear to which possibilities the intersection with $\mathcal{C}$ can lead. Applying the framework of Part II requires to verify for each case the assumption that $\mathcal{K}$ is a connected compact homogeneous manifold. Interestingly, this does not require the group G itself to be compact.

For the second step of the controller, the idea is to define a new cost function as the sum of the cost functions V_k defined in (9.19) and of a configuration cost function V_L as defined in Part II. It seems reasonable to combine those two types of cost functions because the first one drives the agents *towards* $\mathcal{K}$, while the second one governs their positions *along* $\mathcal{K}$.

However, the situation is in fact not so simple. To explicitly write the cost function $V_L : \mathcal{K}^N \to \mathbb{R}$, it is necessary to determine how positions on the particular manifold $\mathcal{K}$ are associated to arbitrary positions g_k on G. It seems that explicitly writing cost functions, deriving control laws along the lines of Section 9.3 and analyzing the convergence of the resulting algorithms requires to particularize the setting and notation to a point where claiming generality becomes questionable. Therefore, more detailed investigation is left for potential particular applications.

O.Q.: In underactuated settings, it is currently an open question if there exists a general explicit control algorithm with convergence proof (maybe for a particular class of systems) to combine left-invariant coordinated motion with the achievement of particular relative positions.

Ex. 9.4.2: steering control with splay state in $SE(2)$: The idea described above is precisely used by [130, 131, 132] to design control algorithms for coordinated motion under steering control on $SE(2)$. In particular, the set $\mathcal{K}$ of positions compatible with steering control and velocity ξ^r is just the isotropy group CM_{ξ^r}, because LIC is equivalent to total coordination in this setting. For rotational motion, CM_{ξ^r} is a circle in $SE(2)$, as explained in Example 8.2.8: the heading θ_k of the planar body entirely determines g_k, because all admissible r_k belong to the same circle in $\mathbb{R}^2$ with angular positions on the circle determined by θ_k such that the agents' velocities are tangent to the circle. Therefore, adding to the sum of V_k a cost function like V_θ defined in (3.10) of Part I, control laws can be designed for the agents to asymptotically achieve a circular motion at fixed speed and with a particular type of configuration where positions of the agents on the circle form e.g. a splay state. More complex configuration cost functions, involving harmonics of the agents' positions on the circle, are used to single out specific configurations. The reader is referred to [130, 131, 132] for a convergence proof. The latter takes advantage of several particular properties of the setting; it appears that generalizing it, if possible at all, would require strong assumptions. $\diamond$

Recapitulation

Part III of the dissertation is devoted to the study of coordinated motion with possibly underactuated agents on Lie groups. Unlike in Parts I and II, the goal is not to reach a particular configuration, but to reach a motion where some variables characterizing the swarm remain constant.

First, left-invariant and right-invariant relative positions are defined on a Lie group. Then left-invariant (resp. right-invariant) coordinated motion is defined as a motion during which the left-invariant (resp. right-invariant) relative positions of the agents remain constant. Also, total coordinated motion is defined as simultaneous left-invariant and right-invariant coordination. It is shown that right-invariant coordination corresponds to equal left-invariant velocities, and left-invariant coordination corresponds to equal right-invariant velocities. This essentially means that coordinated motion on a Lie group is equivalent to synchronization of velocities in a vector space. However, there are two situations where achieving coordination is not straightforward. The first situation is total coordination. It requires synchronization of left-invariant velocities ξ_k^l and of right-invariant velocities ξ_k^r simultaneously, while they are linked through the fundamental position-dependent relation $\xi_k^r = Ad_{g_k}\xi_k^l$. This constrains relative positions compatible with total coordination: for a given velocity ξ, the relative positions must belong to the isotropy subgroup of ξ. The second situation is left-invariant coordination of underactuated agents in a left-invariant control setting (or equivalently, transposing everything to right-invariant). Indeed, then the possible right-invariant velocities depend on the positions of the agents and again, relative positions compatible with coordination are generally restricted (depending on the setting). Implications of these abstract definitions of coordination are illustrated on practical examples $SO(3)$, $SE(2)$ and $SE(3)$ with the interpretation of rigid body motion. Right-invariant coordination corresponds to all agents "drawing translated and rotated versions of the same trajectory"; left-invariant coordination corresponds to motion of the swarm "as a rigid body", with constant (physical) relative positions and/or orientations in body frame.

Control algorithm design to achieve coordinated motion is investigated in an "agreement" setting where the agents communicate with each other along the edges of a specific communication graph and move according to left-invariant first-order dynamics. In this context, right-invariant and fully actuated left-invariant coordination are easily achieved with a consensus algorithm in the Lie algebra. However, the restriction to compatible relative positions for total coordination and underactuated left-invariant coordination requires a two-step strategy: a first step achieves consensus on auxiliary variables to define a desired (mostly right-invariant) velocity, and the second step uses an appropriate cost function to drive the agents to compatible positions. Finally, it is also suggested how to combine the cost functions with those of Part II in order to achieve coordinated motion with a particular consensus configuration (instead of an arbitrary compatible configuration determined by initial conditions).

As an overall conclusion, Part III shows how coordination can be studied in a systematic way once the Lie group geometry of the configuration space is well characterized. The present work proposes few new control laws, but shows how the proposed framework leads to control laws for $SE(2)$ and $SE(3)$ that were proposed in the literature on the basis of intuitive, physical arguments which were difficult to find. It thereby not only suggests how to apply the same framework to other physical settings where agents' motions on a Lie group must be coordinated, but also draws a link with several concepts of mathematical and control systems theory, like the isotropy subgroup or Brockett's double bracket flow.

The original contribution in Part III consists in providing a *unified geometric framework* to study and design algorithms for coordinated motion on Lie groups. Illustration of the geometric concepts on $SE(2)$ and $SE(3)$ leads to existing control laws. For these examples the novelty is thus not in the expression of the algorithms, but in showing that they can be derived in a unified systematic manner with the proper geometric setting.

The original content of Part III is published in [115, 116].

Part IV

Conclusion

The Conclusion is subdivided into four sections. First a summary collects the main points of the dissertation, extracting results from different chapters and sections in order to highlight related observations; for a more linear review of the contributions of the dissertation, the reader is referred to the recapitulations at the end of Parts I, II and III. A second section elaborates on the relevance of the present work for applications. A third section collects the open questions formulated throughout the dissertation and proposes some general directions for future research. The last section reflects on some general lessons to be learnt from this work.

C.1 Summary

Position and Motion coordination. The present dissertation examines two coordination tasks for a swarm of agents evolving on a nonlinear manifold: (i) "position coordination", where the goal is to reach specific configurations (relative positions), including the special case of synchronization (all agents at the same position) and (ii) "motion coordination", where the goal is that agents move with equal velocities and/or maintaining constant relative positions. A last section suggests how both tasks could be combined to get a complete controller for collective motion in formation. For (affinely actuated) points moving in a vector space, position and motion coordination are geometrically equivalent tasks. However, already for agents moving in a vector space with *orientation* being relevant, the overall configuration space is a nonlinear manifold and position or motion coordination are geometrically different.

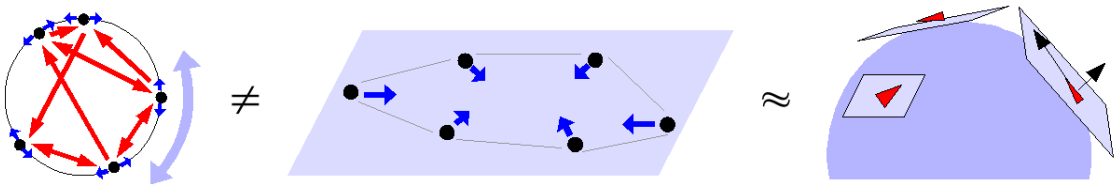


Figure 10.1: Position coordination, e.g. global synchronization, on the circle (left) fundamentally differs from synchronization on vector spaces (middle). Motion coordination on a Lie group, like $SO(3)$ e.g. representing vehicle motion on the sphere (right), can be solved with synchronization on a vector space (middle) only for settings implying no constraints on compatible relative positions; for instance, vector space synchronization is *not* sufficient for *steering controlled vehicles* to achieve motion in formation on the sphere.

Agreement and Symmetry. A basic ingredient of coordination is the *agreement* problem: the individual agents must decide on a common behavior. In absence of external references, the absolute position of the agents has no influence on their behavior, only their relative positions influence the dynamics of the system. Geometrically, this is formalized by considering highly symmetric manifolds, namely connected compact homogeneous manifolds for position coordination and Lie groups for motion coordination, and requiring invariance of the swarm's behavior with respect to a uniform translation of all the agents along the symmetry group of the manifold. This aspect differentiates the present approach from several popular "coordination" algorithms where each agent follows the same external reference.

Geometry. Coordination has scarcely been addressed with a geometric viewpoint in the existing literature. The present work aims at highlighting the importance of the geometry of the configuration space by focusing on specific issues related to coordination on nonlinear manifolds. Locally, position coordination is equivalent on nonlinear manifolds and on vector spaces — it basically reduces to making agents move towards their neighbors to achieve synchronization. In contrast, new issues appear when agents are globally distributed on a manifold. Synchronization algorithms obtained as an extension of vector space consensus algorithms can lead to stable configurations where the agents are actually spread over the entire manifold, or even fail to converge to any fixed configuration. In addition, on compact manifolds like the circle or sphere, it can be meaningful to look for so-called balanced configurations, at the opposite of synchronization, where agents are as far apart as possible.

Motion coordination on nonlinear manifolds differs from vector spaces, even locally. Indeed, on vector spaces, coordinated motion is simply achieved by requiring equal velocities for all the agents. But on a manifold, the tangent planes — and thus the feasible velocities — differ from one position to another. It is therefore necessary to examine meaningful ways to compare velocities at different positions.

Formalization and definition of new concepts. The formalization of position coordination on an embedded connected compact homogeneous manifold starts by defining an easily computable *mean position* of points, called the induced arithmetic mean. The induced arithmetic mean minimizes the chordal distance to all the agents and is locally equivalent to the canonical Karcher mean. It comes down to the projection on the manifold of the arithmetic mean of the position vectors in the embedding vector space. On this basis, several particular configurations are defined for a set of agents on a connected compact homogeneous manifold. Agents have neighbors defined according to some interconnection graph. *Consensus* is a Nash equilibrium where each agent is at the induced arithmetic mean of its neighbors; in such a situation, no individual agent has any incentive to move if its goal is to get as close as possible to its neighbors. Similarly, *anti-consensus* is a situation where no individual agent has to move if its goal is to get as far as possible from the mean of its neighbors. *Balancing* is a configuration where the agents are "distributed in a balanced way", namely such that the induced arithmetic mean of their positions contains all points of the manifold. On a vector space, consensus reduces to synchronization, while anti-consensus and balancing would make no sense.

Coordinated motion is formalized from basic principles of symmetry with respect to left and right translation on a Lie group. This leads to left- and right-invariant relative positions, and allows to compare velocities directly in a common vector space, by translating them to the Lie algebra in a left- or right-invariant way. Meaningful intuitive definitions of coordination, like motion in formation, can be formalized by requiring left- or right-invariant positions to be maintained, which corresponds to having equal right- or left-invariant velocities in the Lie algebra respectively. Imposing left- and right-invariant coordination simultaneously leads to a new concept called *total coordination*; for instance, rigid bodies moving in the plane with total coordination have the same velocity in their body frames and at the same time keep a fixed formation. Total coordination requires the velocity and relative positions to satisfy some compatibility condition: for a given velocity, the relative positions of the agents are restricted to a specific subgroup, like a circle for curved motion in the plane, or a cylinder for curved motion in three dimensions. A similar condition applies for left-invariant coordination of

underactuated agents, which is the formalization of the most frequent setting in the literature. The facts that (i) left-invariant coordination requires to synchronize *right-invariant* velocities, and (ii) in addition to velocity concerns, the relative positions must be driven to a compatible set, explain why the design of coordination control laws on Lie groups is challenging and provide a geometric justification for earlier contributions in the literature. On vector spaces, the Abelian group structure implies that left- and right-invariance are equal, all coordinated motion types reduce to the same and the problem becomes trivial.

Algorithm design and convergence analysis. A central issue in the agreement problem is that every agent does not communicate with every other agent; the limited interconnection among agents is represented by a graph. In order to concentrate on geometric aspects, individual agent dynamics are simplified and practical concerns like time delays, actuator saturation and power consumption are not taken into account. An example of rigid body attitude coordination under mechanical dynamics illustrates how the developed framework can be used in more complex settings; in this context, the dependence of feedback laws on velocity is also briefly discussed and results illustrate the limitations of what can be achieved with only relative velocities.

Two general tools are recurrently used to design control laws for the individual agents. The first method is to formulate objectives in an optimization setting, as the minimizing set of a cost function; the latter is then used for Lyapunov-based controller design, often it is even possible to define gradient algorithms. The second method uses auxiliary variables in order to solve nonlinear agreement problems; in this case, each agent must run an update algorithm for its auxiliary variable and communicate it to its out-neighbors.

Algorithms for position coordination are obtained with a cost function summing the chordal distances between interconnected agents, as a natural extension of vector space consensus algorithms to the circle and further to connected compact homogeneous manifolds. These algorithms and cost function particularize to existing models, like Kuramoto or Vicsek models on the circle and an artificial potential previously proposed for attitude synchronization on $SO(3)$. The first-order controllers locally stabilize synchronization as on vector spaces. Global convergence results require fixed and undirected graphs, for which all stable equilibria are consensus or anti-consensus configurations respectively when the agents are made to gather or to spread. For the complete graph, gathering leads to almost-global synchronization and, although this could not be proved, spreading seems to lead to balancing. Using a dynamic controller which achieves consensus on auxiliary variables that evolve in the embedding vector space, almost-global synchronization and balancing can be obtained for any uniformly connected communication graph. Alternatively, a Gossip Algorithm that randomly selects at most one neighbor to follow at each time step achieves global synchronization with probability 1 for any uniformly connected graph. Another alternative is to use a modified interaction profile such that only synchronization is stable for any connected fixed undirected graph.

Regarding coordinated motion, right-invariant and (essentially) fully actuated left-invariant coordination can be solved by vector space consensus algorithms in a left-invariant setting. For total and underactuated left-invariant coordination, agent positions must be controlled in addition to the velocities. This is done by combining auxiliary variables to define desired velocities and a cost function reflecting the distance of the positions to a compatible set. The resulting general design method can be applied under some conditions and ensures at least local convergence towards the coordinated motion. The use of auxiliary variables again allows

to cope with directed, time-varying, uniformly connected graphs.

Finally, it is suggested how to combine position coordination and motion coordination to get algorithms achieving coordinated motion with particular configurations of the agents.

C.2 Relevance in applications

The present work focuses on the issues of global geometry and invariance in the framework of coordination of multi-agent systems. Although it is unlikely that the assumed simplified setting directly applies to relevant real-world problems, the hope is that the proposed conceptual study might provide two types of insights. From a design point of view, the simple algorithms — or at least their concepts — might be interesting to incorporate in subroutines for solving related real-world problems; this is anyway how most of the theoretically designed controllers are eventually used. The purpose of Part III is clearly to provide a basic tool, helpful in the design of more complex controllers for real applications. From an analysis point of view, the concepts, issues, and examples of possible behavior might help understand some basic mechanisms behind observed collective phenomena, or "order-disorder phase transitions" as they are called in physics. The general geometric formulation of the present work indeed provides a conceptual viewpoint that can be linked to many situations.

Insisting on global geometry and symmetries is necessarily an idealization of reality. In a realistic control framework, external perturbations will almost always break the invariance of the setting with respect to absolute position; it is also rarely the case that the operational objective is strictly independent of absolute positions. For instance, satellites in space are subject to many external torques (gravitational pull of other bodies, solar wind,...) and, in rare occasions when their science objective does not require them to point in a particular direction (e.g. thinking of assembly [55, 86]), their orientation and position with respect to the Earth (e.g. communication, arrival of new parts), the Sun (temperature and energy requirements),... is still relevant to some extent. However, several reasons still make it important to be aware of the issues raised in Parts I and II of the present work.

- For small symmetry-breaking perturbations, the specific phenomena arising in the global behavior of synchronization algorithms on nonlinear manifolds persist. This means among others that using some preferred reference point in the agreement process, it must have a strong enough weight with respect to the attraction/repulsion of neighboring agents for the convergence to become vector space -like.
- The present framework considers synchronization algorithms on manifolds for *arbitrary, globally distributed* initial conditions and, only considering relative positions of interconnected agents, requires *limited information exchange*. It thereby does not necessarily address the most critical questions for real operational modes of a swarm — for instance, the main concern for satellite interferometers is to reach the required *accuracy* near equilibrium, see [9]. However, it increases robustness, since synchronization does not rely on permanent communication of a common reference, nor on the health of a potential leader, nor on the assumption that the agents are initially close to the objective. This may be useful for deployment or recovery modes, where there is no guarantee on initial conditions and communication links may fail.
- The assumed invariance allows to stabilize a formation without interfering with its absolute motion. This leaves the freedom to design an absolute motion independently of the coordination process, focusing on specific task-oriented objectives. When absolute

position is barely relevant, it may be possible for instance to lower power consumption in a distributed way; this probably requires some adaptation of the coordination algorithms, but the methodology could be the same as in Part II. In general, achieving coordination in an invariant setting, independently of an absolute reference, provides a swarm that behaves more like a single body than like a set of individuals.

- Invariance of the algorithms is important for computational tasks like data analysis with no a priori knowledge, in order to introduce no bias in the results.

It is difficult to assess to what extent useful formations for practical purposes can be built with the formalism of consensus and/or anti-consensus configurations. A hint in this direction could be a result in [131] which stabilizes formations on the circle by combining several cost functions of the consensus/balancing type involving harmonics of the agents' relative positions; there is however currently no conclusive extension of this approach to more general manifolds. The definition of consensus and the corresponding behavior of the agents are in fact intuitive and simple enough to be plausible primitives for swarm behavior on manifolds.

The developments of Part II might prove interesting in a computational framework. The induced arithmetic mean is particularly simple to compute in general, as testified by explicit expressions for $SO(n)$ and the Grassmann manifolds. The number of points just determines a number of additions to perform before applying *a single* computationally more heavy projection operation; this ensures excellent scalability. The question is then rather how relevant this particular definition of mean is for a given application. An important computational task that may benefit from the induced arithmetic mean, decentralized gradient algorithms and related developments in Part II, is *data clustering* on manifolds. Given a large set of points on a state space, the task of clustering is to partition the points into a given number of sets and define a new "ambassador" point for each set in such a way that the ambassadors alone are as representative as possible of all the initial data. A plausible ambassador for a set is the mean of its points, which draws the link to the induced arithmetic mean. A corresponding k-means algorithm has recently been proposed for Grassmann manifolds in [45].

It is still a question for further investigation to identify applications for consensus, anti-consensus and balancing configurations in the computational framework. This is certainly partly due to the fact that such configurations are not completely characterized in a practically exploitable way for other manifolds than the circle. Questions involving "optimally distributing" points on manifolds are ubiquitous basic computational tasks; for instance, the *packing* problem seeks to arrange a set of points on a state space in such a way that the minimal distance between any two points is maximized. The link between such applications and consensus, anti-consensus or balancing as defined in the present work still has to be established. Investigating the applications may also help to better characterize consensus configurations. In any case, moving agents towards or away from the mean of their neighbors on a manifold can be a heuristic method to solve optimal distribution tasks like packing, provided that the neighbors are defined in an appropriate, probably state-dependent way. Packing requires to distribute points on the whole manifold, and hence illustrates how issues related to global geometry can become important in the computational context.

The developments about coordinated motion in Part III are directly based on the Lie group geometry of the setting. As can be seen from examples, the Lie group setting of the configuration space in fact formalizes intuitive practical situations; note the importance to

take into account not only positions, but also orientations of the vehicles in the model.

The importance of global geometry is not due to a global distribution of the agents on the configuration space; in fact, they may be very close to each other. However the *moving* swarm explores the manifold *in time*. Invariance with respect to translations on the Lie group must therefore be seen as a requirement that the swarm's behavior remains constant in time; in this context the *exact* Lie group symmetry is a *simplifying* assumption. Like for the application of Parts I and II to formation control, neglecting any external reference can be interesting in order to decouple the coordination process from the actual trajectory of the coordinated swarm.

C.3 Leads for future work

Future research related to the present work can be subdivided into two categories: (i) filling remaining gaps (open questions) in the canvas of the dissertation and (ii) extending its concepts in various directions, some of which may require substantial efforts; application-oriented developments are not repeated here since they are discussed in the previous section.

The following open questions have been identified throughout the dissertation.

- (Chapter 3) How can S^1-synchronizing graphs — i.e. interconnection graphs for which synchronization is the only stable equilibrium of classical consensus algorithm (3.12) on the circle — be characterized with simple graph properties ?
  (Chapter 5) For which graphs does the consensus algorithm of Section 6.1 almost-globally lead to synchronization on general manifolds ?

- (Chapter 3) What are the possible behaviors of the Vicsek model (3.8),(3.9) for various initial conditions ?

- (Chapter 4) What is the expected global convergence rate of the Gossip Algorithms towards synchronization, as a function of interconnection graph and link choice probability distribution ? How should the probability distribution for random link selection be chosen in order to maximize the convergence rate ?

- (Chapter 5) For which graphs can all possible consensus and/or anti-consensus configurations be easily characterized (maybe depending on particular manifolds) ? Which of these configurations are stable under the gradient algorithm of Section 6.1 ? In particular, how frequent are stable anti-consensus configurations for the complete graph which differ from balancing ?

- (Chapter 5) How can the sophisticated cost function of [131], involving harmonics of the agents' positions on the circle to stabilize specific balanced configurations, be extended to general manifolds in a meaningful way ?

- (Chapter 6) How can the auxiliary variables, required by the algorithms of Section 6.2.1, be transmitted in "agent-centered" coordinates — i.e. as arrays of scalars without requiring the help of a common external reference frame — on general connected compact homogeneous manifolds ?

- (Chapter 6) Is there a way to design a synchronization and/or balancing algorithm with similar performance as those proposed in Section 6.2.1, but which makes better use of auxiliary variables: either (i) requiring auxiliary variables of smaller dimension or (ii) more to the point, not requiring explicit *communication* of auxiliary variables among agents, or (iii) requiring communication of static additional information, without a separate dynamic algorithm for the auxiliary variables ?
- (Chapter 6) Is it possible to propose a general modified interaction profile among agents on connected compact homogeneous manifolds, in a synchronization algorithm like the one of Section 6.1, such that the only stable configuration is synchronization for any connected undirected fixed $\mathbb{G}$?
- (Chapter 6) Can asymptotic synchronization with probability 1 be proved on any connected compact homogeneous manifold for the undirected Gossip Algorithm ?
- (Chapter 7) In a setting of rigid body attitude control with mechanical dynamics, how is it possible to combine the extensions of Section 7.3.2 — using only *relative* angular velocities $(Q_j\omega_j - Q_k\omega_k)$ — and of Section 7.3.3 — using a consensus algorithm on auxiliary variables — to obtain a control algorithm that (i) achieves almost global attitude synchronization for varying and directed interconnection graphs and (ii) at the same time, allows any synchronized free rigid body motion ?
- (Chapter 8) Can the framework of total coordination be used to indirectly define practically relevant configurations ?
- (Chapter 9) Is there a general convergence result for the adaptation to *underactuated* agents of the total coordination algorithms on Lie groups of Section 9.2.2 — i.e. characterizing the convergence properties of the resulting asymptotically autonomous systems whose asymptotic dynamics are the projection of a gradient system onto the range of an underactuated controller ?
- (Chapter 9) Is there an easy general method for the design of left-invariant consensus algorithms to properly agree on a desired right-invariant velocity for *underactuated* left-invariant coordination ?
- (Chapter 9) Is there a general explicit control algorithm with convergence proof (maybe for a particular class of systems) to combine left-invariant coordinated motion control with the achievement of particular relative positions in underactuated settings ?

Some of the above questions lead to more fundamental issues. Several future directions can be imagined for the present work; the following discussion is not meant to be exhaustive.

An important theoretical and practical issue is the consideration of control cost. The algorithms proposed for synchronization in this dissertation maintain invariance of the swarm with respect to the place of meeting. One might then ask the question: what is the best meeting place such that total **motion power consumption** in the swarm is minimized, or such that the maximal power consumed by any individual agent is minimized ? The question becomes more relevant, also more challenging and theoretically interesting, in a framework with mechanical dynamics like the Euler equations for rigid body rotation. A second important aspect of power consumption is the required **communication power or bandwidth**.

Indeed, given the difficulties observed in the present dissertation with reduced numbers of communication links among agents, and e.g. the increase of exchanged information packages when a global convergence problem is solved using auxiliary variables, it could be interesting to examine more closely the tradeoff between achievable performance and requirements of connectivity, bandwidth and power on the communication system.

This last issue also takes importance in settings with **state-dependent communication links**, because in practical situations, power consumption and the possibility to establish a link are likely to depend on relative positions of the agents. More generally, the behavior of systems where the presence or absence of a communication link between agents depends on their relative states is scarcely studied in the literature: most existing results assume, as in the present dissertation, that the interconnection graph satisfies appropriate properties. From a design viewpoint, some authors propose sophisticated conservative solutions, where a major objective for the motion design is to maintain small enough distances ensuring that communication links are kept active [30, 101]. However, from an analysis viewpoint, it is expected that the interplay between graph dynamics and state dynamics could give rise to a rich variety of behaviors in simple systems like the Vicsek model [148]. The identification of specific mechanisms related to coordination with state-dependent interaction graphs could open an interesting viewpoint on the study of interacting particles. See [3, 50, 14] for preliminary work in this direction. The investigation of appropriate tools to study coordination with state-dependent graphs may also foster interesting developments in relation with the theory of hybrid systems (i.e. systems mixing continuous-time dynamics with discrete transition maps where the dynamics change).

Another issue is related to the **use of auxiliary information** in addition to just relative positions on the manifold. A solution for global synchronization in the present work takes a rather drastic approach by appending to the position controller a whole separate consensus algorithm in the embedding vector space. Carrying this idea to the extreme, one could imagine to let the agents first run an algorithm to elect a leader, and then follow that leader's decision, relayed through other agents if there is no direct communication link. More moderate approaches could also convey interesting information to individual agents in order to enhance their convergence properties. For instance, an agent computing the induced arithmetic mean of its neighbors can also consider the distance from the centroid in the embedding space to the manifold, or to a balanced state, in order to take a decision. Adding such information to the communicated information package could provide the agents with some "confidence weights" about the position and motion of their neighbors, and e.g. the latter could be momentarily discarded, or conversely favored, depending on their confidence levels. It might be interesting to study which kinds of local statistics are interesting to propagate in such a way through the swarm, how these statistics can best be used by the receiving agents, and to what kind of behavior they can lead. Additional statistics about a swarm's configuration may be of interest in other problems, involving for instance clustering methods on manifolds.

A more exploratory direction would be to link the setting of the present dissertation to the modeling of interacting physical particles. The issues of invariance and global geometry take primary importance in relation with the description of physical systems. A first new path would be to establish a **continuous fluid flow** limit for an infinite number of agents, as is done in [50] on the real line. This would yield partial differential equations on manifolds and maybe allow to draw interesting conclusions for the latter. Communication among agents will then probably be modeled in a state-dependent way; this would even lead to integro-differential equations if a finite "communication domain" is considered for each parti-

cle. Such an extension could be inspired by the seminal work of Professor P.-L. Lions, whose recent developments about mean field games on vector spaces seem to have a basis similar to consensus behaviors. The latter work also introduces **stochasticity** in the agents' behavior, drawing a direct connection to statistical mechanics. The brief study of the Gossip Algorithm in the present dissertation illustrates how the introduction of randomness can affect overall behavior of a swarm on a nonlinear manifold. A link with the study of **coupled quantum systems** also seems interesting to investigate.

C.4 General lessons

The present section takes a step back from the particular problems studied in this dissertation, in order to briefly examine more general considerations.

A first general observation concerns the properties of manifolds. The literature about nonlinear manifolds covers a whole branch of mathematics known as the differential geometry. This theory defines manifolds as objects that, locally, can be seen as the image of a subset of a vector space by a differentiable invertible mapping. It describes how to formally characterize tangents, scalar products ("metrics"), first- and higher-order derivatives of functions on manifolds, many specific objects like geodesics, parallel transport,... . All these quantities can be explicitly computed using local coordinates thanks to the local mapping to a vector space. This is very useful for instance to study local optimization algorithms (see e.g. [2]) or Einstein's theory of general relativity, to which it was initially mainly connected (see e.g. [127]). However, it seems that there exist tasks, like the coordination problem studied in the present dissertation, where considering manifolds *locally* is insufficient. Indeed, the *global* geometry (more precisely, topology) can lead to important new phenomena, like the ones described in Section 3.4 for the circle, that would be absent from a local treatment. Thus *nonlinear manifolds are really more fundamentally different from vector spaces when their global geometry is of importance* than what one could be led to believe from the vast theory of differential geometry. A detail in this respect, mainly observed in Part II (for the Grassmann manifolds), is that using representations which *embed* the manifold in a vector space are handy when discussing properties of global geometry, although they can be of much higher dimension than other representations (like the viewpoint of quotient manifolds).

The root of most difficulties in the coordination problems of the present dissertation is the requirement of invariance: this is the basic reason why a global agreement problem must be solved by the agents. In this sense, the present dissertation highlights the *importance of symmetry and geometric invariance*, which give rise to specific problems and phenomena that are absent from non-symmetric settings. An example of this contrast is the invariant study in Part II of this dissertation as opposed to non-invariant settings for synchronization in the presence of reference tracking (see corresponding literature in the Introduction of Part II and discussions at various places in this dissertation). When investigating basic coordination mechanisms in natural systems, it is important to take invariance into account, because the physical world is fundamentally symmetric with respect to several transformations.

Symmetry on non-Euclidean spaces can lead to "annoying" behavior in a swarm of agents, like e.g. the impossibility to define a unique mean for agents uniformly distributed on the circle. In order to reach synchronization in spite of these problems, it can be necessary to

"break the symmetry" in some way. The present dissertation introduces several mechanisms for avoiding symmetry-related problems without completely losing the fundamental invariance of the dynamical system, including mainly the following ones.

- The worst situation in a swarm is when agent positions and the interconnection graph have "the same symmetry", like for a splay state on the circle with an undirected ring graph. The symmetry can be broken by significantly changing the relative importance given by an agent to its different neighbors; this can be implemented by redefining the edges of the interconnection graph. *Randomly* switching the links in the swarm, as first proposed in Section 4.2, is a solution that properly maintains the invariance of the initial setting. It *ultimately irreversibly swaps the system into the desired basin of attraction.* This kind of approach might be of interest in connection with the naturally probabilistic setting of quantum mechanics. In this context, it is also important that "less can be more", since global convergence is in fact obtained by *suppressing* connections between agents, e.g. cutting a ring to a path. Convergence is thus not just a function of the quantity of exchanged information, but depends on the structure of the network.
- The symmetry of a nonlinear manifold can lead to spurious stable local equilibria of high symmetry. This can be overcome by endowing the agents with auxiliary variables that evolve in a vector space, as first proposed in Section 4.3. This approach is probably best suited in engineering problems. Its relevance for describing natural systems is doubtful, since agents would have to exchange messages containing the values of their auxiliary variables. However, it underlines that the presence of hidden variables evolving in a vector space can break symmetric situations for related variables evolving in a nonlinear manifold.

A more general lesson learnt from the approach of the present dissertation is that *emphasizing a geometric formulation allows to make links between many related problems* that are maybe not apparent when more particular formulations are used. Indeed, in the present case, it comes as a nice surprise to see that many coordination problems can be tied together. Among others:

- The Kuramoto model, the Vicsek model and (less closely) the Hopfield model proposed to describe the collective behavior of multiple autonomous agents are all related to each other; they can in fact be seen as natural extensions to the circle (or the "0-dimensional circle" $\{-1, 1\}$) of the more basic linear consensus algorithm on vector spaces.
- A definition of "mean position" on manifolds, natural consensus algorithms, their local equilibria, and problems of distributing ("balancing") agents on manifolds are naturally derived from each other in a general geometric language.
- Different algorithms that were previously intuitively derived for coordinated motion of rigid bodies (see [126, 131]) are placed in a common framework of coordinated motion on Lie groups. A general geometric design method is proposed which explains how (and why) to design such algorithms for coordinated motion on Lie groups, retrieving the intuitively derived algorithms as particular cases. The developed theory, especially regarding total coordination, makes use of numerous classical objects of Lie group theory, involving e.g. adjoint orbits, conjugation classes and isotropy subgroups.
- A link is established between the latter algorithms and the double-bracket flow studied by R. Brockett (see [19]).

Highlighting all these links between different problems might even be seen as part of the contribution of the present work.

Finally, an important insight gained by working on the present subject is to *never underestimate the power of simple ideas*. A personal experience during this work was to realize how a basic idea on the simplest example can sometimes open up a new viewpoint, throw important light on the matter and extend to more general cases; even then, the full implications of the new fact might be long to establish, linking up with other subjects and open/known problems, potentially opening directions for further research.

Appendix

A.1 Proof of Proposition 3.3.5

Two steps are used to derive a sufficient bound on the value of β ensuring convergence for synchronous operation of (3.6).

Step 1: Consider a first-order Euler discretization of (3.12) with $a_{jk} \in \{0, 1\}$ in the form

$$\theta_k(t+1) = \theta_k(t) - \alpha\, T\, \frac{\partial V_\theta}{\partial \theta_k}\,, \quad \alpha > 0\,, \tag{11.30}$$

where T is the discretization step. The variation of V_θ between two time steps is

$$\begin{aligned} V_\theta(t+1) - V_\theta(t) \;\; &= \;\; -\alpha\, T\, \|\nabla_\theta V_\theta\|^2 \\ &\quad + \tfrac{1}{2!}\nabla_\theta^2 V_\theta \prod_{m=1}^{2} \times_m (\alpha\, T\, \nabla_\theta V_\theta) - \tfrac{1}{3!}\nabla_\theta^3 V_\theta \prod_{m=1}^{3} \times_m (\alpha\, T\, \nabla_\theta V_\theta) + ... \end{aligned} \tag{11.31}$$

where ∇_θ denotes the "overall gradient" operator whose application to a function f of $(\theta_1, \theta_2, ..., \theta_N)$ yields N components $(\frac{\partial f}{\partial \theta_1}, \frac{\partial f}{\partial \theta_2}, ..., \frac{\partial f}{\partial \theta_N})$, and $\times_m$ denotes tensor multiplication along dimension m. According to (11.31), V_θ is non-increasing at least when the first term on the right-hand side is dominant. To satisfy this condition, a bound is imposed on the value of $\alpha\, T$. For this purpose, rewrite

$$V_\theta = \sum_{k=1}^{N} \sum_{j \rightsquigarrow k} (1 - \cos(\theta_j - \theta_k))$$

and consider its derivatives with respect to θ_l. Each edge in the graph appears in the tensor $\nabla_\theta^n V_\theta$ exactly $\sum_{k=0}^{n} (C_n^k\, 1^k\, 1^{n-k}) = 2^n$ times, where $C_n^k := \frac{n!}{k!(n-k)!}$. Then by giving the maximal value 1 to all appearing sines and cosines, one concludes that the sum of the absolute values of all elements in $\nabla_\theta^n V_\theta$ is smaller than $d_{sum}\, 2^n$, i.e.

$$|\nabla_\theta^n V_\theta| \prod_{m=1}^{n} \times_m \mathbf{1}_N \leq d_{sum}\, 2^n\,.$$

For any vector $x \in \mathbb{R}^N$ and any tensor $A_n \in \mathbb{R}^{N \times N \times ... N \text{ (n times)}}$ of degree $n \geq 2$, it holds

$$\left| A_n \prod_{m=1}^{n} \times_m x \right| \leq |A_n| \prod_{m=1}^{2} \times_m \mathbf{1}_N\, x^T x \prod_{m=3}^{n} \times_m |x| \leq |A_n| \prod_{m=1}^{n} \times_m \mathbf{1}_N\, x^T x \left(\max_{k \in \mathcal{V}} |x_k| \right)^{n-2}\,.$$

In particular, replacing x by $(\alpha\, T \nabla_\theta V_\theta)$ and A_n by $\nabla_\theta^n V_\theta$, the higher order terms in (11.31) are bounded by

$$\begin{aligned} \frac{\left|(V_\theta(t+1) - V_\theta(t) + \alpha T\, \|\nabla_\theta V_\theta\|^2\right|}{\alpha T\, \|\nabla_\theta V_\theta\|^2} \;\; &\leq \;\; \frac{\sum_{n=2}^{+\infty} \frac{1}{n!} \left| A_n \prod_{m=1}^{n} \times_m x \right|}{\alpha T\, \|\nabla_\theta V_\theta\|^2} \\ &\leq \;\; \alpha T\, d_{sum} \sum_{n=2}^{+\infty} \frac{1}{n!} 2^n\, (2\alpha T d_{max})^{n-2} \leq \frac{d_{sum}}{M\, d_{max}} \sum_{n=2}^{+\infty} \frac{M^n}{n!}\,, \end{aligned}$$

where $M = 4\alpha T\, d_{max}$. This implies that the first term of (11.31) will be dominant if

$$\frac{e^M - 1}{M} \leq 1 + \frac{d_{max}}{d_{sum}}$$

which may be solved numerically to produce a sufficient convergence condition $M \leq M^*$. This in turn fixes a sufficient bound on αT for a given topology, which through (11.30) leads to the following equivalent requirement on the motion of agent k between two time steps:

$$|\theta_k(t+1) - \theta_k(t)| \leq \frac{M^*}{4d_{max}} \left| \frac{\partial V_\theta}{\partial \theta_k} \right| . \tag{11.32}$$

Step 2: The second step is to connect (11.30) to the actual algorithm (3.6). After some elementary geometrical observations, one verifies that for $\beta > d_k$, the distance $\theta_k(t+1)-\theta_k(t)$ travelled along the circle when applying (3.6) satisfies

$$|\theta_k(t+1) - \theta_k(t)| \leq \frac{1}{2d_k} \left| \frac{\partial V_\theta}{\partial \theta_k} \right| \frac{d_k}{\beta - d_k} .$$

Comparing with (11.32) finally leads to the condition $\beta \geq \frac{2d_{max}}{M^*} + d_k \ \forall k \in \mathcal{V}$ and taking the maximum over all $k \in \mathcal{V}$ yields the condition of Proposition 3.3.5.

A.2 Lemma for the proof of Proposition 6.1.5

Lemma A.1: *If $g(Q) = Q^T B - B^T Q$ with $Q \in SO(n)$ and $B \in \mathbb{R}^{n\times n}$, then $g(Q) = 0$ if and only if $Q = UHJH^T$, where $B = UR$ is a polar decomposition of B, the columns of H contain orthonormalized eigenvectors of R and*

$$J = \begin{pmatrix} -I_l & 0 \\ 0 & I_{n-l} \end{pmatrix}, \qquad \begin{array}{ll} l \text{ even} & \text{if } \det(U) > 0 \\ l \text{ odd} & \text{if } \det(U) < 0 \end{array} .$$

Proof: All matrices Q of the given form ensure that $Q^T B$ is symmetric. The following constructive proof shows that this is the only possible form.

Since $U^T B = R$ is symmetric with U orthogonal, the problem is to find all orthogonal matrices $T = U^T Q$ such that $S := T^T R$ is symmetric and $\det(T) = \det(U)$. Take a basis of $\mathbb{R}^n$ consisting of eigenvectors diagonalizing R with its eigenvalues placed in decreasing order $\lambda_1 \geq \lambda_2 ... \geq \lambda_n \geq 0$; denote by H^* the orthogonal matrix chracterizing this basis (i.e. whose columns contain the basis vectors). The following shows that T is diagonal in that basis. Then orthogonality of T imposes values 1 or -1 on the diagonal, the number l of -1 being compatible with $\det(T) = \det(U)$. The final form follows by returning to the original basis and reordering the eigenvectors such that those corresponding to -1 are in the first columns.

The j^{th} column of S is simply the j^{th} column of T multiplied by λ_j. Therefore:

- If $\lambda_k = \lambda_j$ for some $j \neq k$, then H^* can be adapted such that the corresponding submatrix $T(j:k,\, j:k) :=$ intersection of rows j to k and columns j to k of T, is diagonal.
- Define p such that $\lambda_{p+1} = 0$ and $\lambda_p \neq 0$; then S symmetric implies $T(n-p:n,\, 1:p) = 0$. Since $T(n-p:n,\, n-p:n)$ is diagonal from the previous item, only diagonal elements are nonzero in the last $n-p$ rows of T. Rows and columns of T being normalized, $T(1:p,\, n-p:n) = 0$.

- Consider $k_- \leq p$, and k_+ the smallest index such that $\lambda_{k_+} < \lambda_{k_-}$. It holds

$$\begin{array}{l} \sum_j \ T^2_{k_- j} = \sum_j \ T^2_{j k_-} = 1 \text{ (orthogonality) and} \\ \sum_j \ S^2_{k_- j} = \sum_j \ S^2_{j k_-} \qquad \text{(symmetry).} \end{array} \tag{11.33}$$

  Start with $k_- = 1$ and assume $\lambda_{k_+} > 0$. Then (11.33) can only be satisfied if $T_{jm} = T_{mj} = 0 \ \forall j \geq k_+$ and $\forall m \in [k_-, k_+)$. The first item further implies $T_{jm} = T_{mj} = 0$ $\forall j \neq m$ and $\forall m \in [k_-, k_+)$. This argument is repeated by defining the new k_- as being the previous k_+ until $\lambda_{k_+} = 0$ (second item) or $\lambda_{k_-} = \lambda_n > 0$. This leaves T diagonal. □

A.3 Bound for the proof of Proposition 7.2.2 (a)

Property: *In the framework of the proof of Proposition 7.2.2(a), the condition*

$$\mu > (\sqrt{6}\alpha d_k^{(i)} J_{k1} + \tfrac{3J_{k1}+J_{ki}}{2}\|\omega_0\|) \quad \forall k \in \mathcal{V}$$

ensures that, with some $\lambda_1 > 0$, $\lambda_2 > 0$ *and* $\lambda_3 > 0$,

$$\tfrac{d}{dt}W \leq \sum_{k=1}^{N} -\lambda_1 \|\omega_k^{(r)} - \omega_k^{(d)}\|^2 - \lambda_2 \|\omega_k^{(r)}\|^2 - \lambda_3 \|\omega_k^{(d)}\|^2 \, .$$

Proof: Consider expression (7.20). Write

$$(J_k\omega_k) \wedge \omega_k = (J_k(\omega_k^{(r)} + \omega_0)) \wedge (\omega_k^{(r)} + \omega_0)$$

and use $(J_k\omega_0) \wedge \omega_0 = 0$ (ω_0 being aligned with a principal axis) to compute

$$\begin{array}{rl} \left(\omega_k^{(r)} - \omega_k^{(d)}\right)^T \left((J_k(\omega_k^{(r)} + \omega_0)) \wedge (\omega_k^{(r)} + \omega_0)\right) = & \\ - \quad (\omega_k^{(d)})^T((J_k\omega_k^{(r)}) \wedge \omega_k^{(r)}) - \omega_0^T((J_k\omega_k^{(r)}) \wedge \omega_k^{(r)}) & \\ - \quad (\omega_k^{(d)})^T((J_k\omega_k^{(r)}) \wedge \omega_0) - (\omega_k^{(d)})^T((J_k\omega_0) \wedge \omega_k^{(r)}) \, . & \end{array} \tag{11.34}$$

The sum of the first two terms is bounded by $J_{k1}\,(\sqrt{6}\alpha d_k^{(i)} + \|\omega_0\|)\,\|\omega_k^{(r)}\|^2$ because

$$\|\omega_k^{(d)}\| = \tfrac{\alpha}{\sqrt{2}} \left\| \textstyle\sum_j \ a_{jk}\,(Q_k^T Q_j - Q_j^T Q_k) \right\| < \sqrt{6}\alpha d_k^{(i)} \, .$$

The sum of the last two terms is bounded by

$$(J_{k1} + J_{ki})\|\omega_0\|\|\omega_k^{(r)}\|\|\omega_k^{(d)}\| \ = \ -\tfrac{J_{k1}+J_{ki}}{2}\|\omega_0\| \left((\|\omega_k^{(r)}\| - \|\omega_k^{(d)}\|)^2 - \|\omega_k^{(d)}\|^2 - \|\omega_k^{(r)}\|^2 \right) .$$

Hence, overall

$$\begin{array}{rcl} \tfrac{d}{dt}W & \leq & \sum_k \ -(\mu\kappa - \frac{1}{\alpha})\|\omega_k^{(r)} - \omega_k^{(d)}\|^2 \ - \left(\frac{1}{\alpha} - \kappa(\sqrt{6}\alpha d_k^{(i)} J_{k1} + \frac{3J_{k1}+J_{ki}}{2}\|\omega_0\|)\right) \|\omega_k^{(r)}\|^2 \\ & & - \kappa\, \frac{J_{k1}+J_{ki}}{2}\,\|\omega_0\|\,(\|\omega_k^{(r)}\| - \|\omega_k^{(d)}\|)^2 \ - \left(\frac{1}{\alpha} - \kappa(\frac{J_{k1}+J_{ki}}{2}\|\omega_0\|)\right) \|\omega_k^{(d)}\|^2 . \end{array}$$

All the terms have negative coefficients if $\kappa > \frac{1}{\mu\alpha}$ and $\kappa(\sqrt{6}\alpha d_k^{(i)} J_{k1} + \frac{3J_{k1}+J_{ki}}{2}\|\omega_0\|) < \frac{1}{\alpha}$. If the condition in the statement on μ is satisfied, then $\kappa > 0$ in the Lyapunov function can be chosen to satisfy these conditions ensuring $\frac{d}{dt}W \leq 0$. □

A.4 Details in the proof of Proposition 7.3.2 (b)

First property: *In the context of the proof of Proposition 7.3.2 (b), given a neighborhood* $W \ni (Q_1, Q_2, ..., Q_N, \omega_1, \omega_2, ..., \omega_N)$ *of* S_{M^*}*, there exist* $|\sigma_1|$ *and a neighborhood* U *of* S_{M^*} *such that starting in* U *implies staying in* W *if* $|\sigma| > |\sigma_1|$.

Proof: Defining $(d_{jk})^2 := 3 - \text{trace}(Q_k^T Q_j)$ to represent the chordal distance (see Chapter 5) between $Q_j \in SO(3)$ and $Q_k \in SO(3)$, let $W = \{(Q_1, Q_2, ..., Q_N, \omega_1, \omega_2, ..., \omega_N) : (d_{jk})^2 < \varepsilon \ \forall j, k \in \mathcal{V}$ and $\|M - M^*\| < \kappa\}$. If $|E(\mathbb{G})|$ is the number of edges in $\mathbb{G}$, then condition

$$\tfrac{1}{2} \textstyle\sum_k \sum_{j \rightsquigarrow k} \ (3 - \text{trace}(Q_k(t)^T Q_j(t))) = 3|E(\mathbb{G})| - V(t)/\sigma < \varepsilon$$

(the factor $\frac{1}{2}$ comes from counting each edge twice) is sufficient for a solution starting with $\|M - M^*\| < \kappa$ to be in W at time t. For $t \geq 0$, since H decreases, $T(t) + V(t) \leq T(0) + V(0)$ so $V(t) - V(0) \leq T(0) - T(t) \leq T(0)$. Hence if $|\sigma| > |\sigma_1|$, then $(V(0) - V(t))/\sigma \leq T(0)/|\sigma_1|$ and so

$$3|E(\mathbb{G})| - V(t)/\sigma \leq (3|E(\mathbb{G})| - V(0)/\sigma) + T(0)/|\sigma_1| \ .$$

Now it suffices to take appropriate initial conditions. Choose a neighborhood $U_1 \subseteq W$ of S_{M^*} such that $\max_k \|Q_k J \omega_k - \frac{M^*}{N}\| < \frac{\beta}{N}\|M^*\|$, for some $\beta > 0$. Initial conditions in U_1 imply $T(0) < \frac{\|M^*\|^2}{2J_3 N}(1+\beta)^2$. Then (assuming the value of M^* unknown) taking $|\sigma_1| > \frac{M_{max}^2}{\varepsilon J_3 N}(1+\beta)^2$ ensures $T(0)/|\sigma_1| < \frac{\varepsilon}{2}$. Also define U_2 to be the set in state space where $3|E(\mathbb{G})| - V/\sigma < \varepsilon/2$. Then with initial conditions in $U = U_1 \cap U_2$, the system stays in W for $t \geq 0$. □

Second property: *In the context of the proof of Proposition 7.3.2 (b), consider solutions of (7.3),(7.4) with identical* $Q_k \omega_k$ *and* $u_k = u_k^{(P)}$ *as defined in (7.22). There exist* $|\sigma_2|$ *and a neighborhood* W_1 *of* S_{M^*} *such that for* $|\sigma| > |\sigma_2|$*, solutions that stay in* W_1 *are necessarily in* S_{M_0} *for some (initial=final) angular momentum* M_0.

Proof: Define $(d_{jk})^2 := 3 - \text{trace}(Q_k^T Q_j)$ like for the first property. Denote the final common velocity (guaranteed by part (a) of the Proposition) by $Q_k \omega_k =: \Omega(t) \ \forall k \in \mathcal{V}$; it holds $\|\Omega\| \leq \frac{\|M_0\|}{N J_3}$. Since $\frac{d}{dt}(Q_k^T Q_j) = 0$ when $Q_k \omega_k = Q_j \omega_j$, the time derivative of $\Omega = Q_k \omega_k$ along solutions of the closed-loop system is

$$\tfrac{d}{dt}\Omega = -\sigma \ Q_k J^{-1} \textstyle\sum_{l \rightsquigarrow k} [Q_k^T Q_l - Q_l^T Q_k]^\vee \ + \ Q_k J^{-1} Q_k^T [Q_k J Q_k^T \Omega]^\wedge \, \Omega \qquad (11.35)$$

which holds $\forall k \in \mathcal{V}$. Denoting the first and second terms of the right side of (11.35) by $(11.35a)_k$ and $(11.35b)_k$ respectively,

$$\|(11.35a)_k - (11.35a)_j\|^2 = \|(11.35b)_k - (11.35b)_j\|^2 \qquad (11.36)$$

$\forall k, j \in \mathcal{V}$. The right side of (11.36) can be bounded by

$$\|(11.35b)_k - (11.35b)_j\|^2 \leq \tfrac{16 J_1^2}{J_3^2} \|\Omega\|^4 \, (d_{jk})^2 \, .$$

using worst-case products, norms of sums transformed into sums of norms and other classical bounds. Thus the same bound must hold for the left side of (11.36),

$$\|(11.35a)_k - (11.35a)_j\|^2 \leq \tfrac{16 J_1^2}{J_3^2} \|\Omega\|^4 \, (d_{jk})^2 \, .$$

Summing the last condition over all j, k, using conservation of M and linearizing leads to (with the same type of calculations as for the first bound)

$$\frac{2\sigma^2\lambda_2^3}{J_1^2}(d_{max}^2 + \mathcal{O}(\, d_{max}^4)) \leq \frac{16J_1^2\,|E(\mathbb{G})|\,\|M^*\|^4}{J_3^6 N^2}\, d_{max}^2 + \mathcal{O}(d_{max}^3) \tag{11.37}$$

where $\lambda_2 > 0$ is the second-smallest eigenvalue of the Laplacian L of $\mathbb{G}$, notation $|E(\mathbb{G})|$ denotes the number of edges in $\mathbb{G}$, and d_{max}^2 denotes the maximal value of $(d_{jk})^2$ among all pairs of connected agents. Choosing a neighborhood $W_1 = \{(Q_1, Q_2, ..., Q_N, \omega_1, \omega_2, ..., \omega_N) : (d_{jk})^2 < \varepsilon \ \forall j, k \in \mathcal{V}$ and $\|M - M^*\| < \kappa\}$ of synchronization such that higher-order terms represent less than $\gamma_1 < 1$ and $\gamma_2 < 1$ respectively on the left and right side of (11.37), it becomes

$$d_{max}^2 \leq \frac{8J_1^4\,|E(\mathbb{G})|}{(1-\gamma_1)(1-\gamma_2)J_3^6\lambda_2^3}\frac{\|M^*\|^4}{\sigma^2}\, d_{max}^2\,. \tag{11.38}$$

Taking $\sigma^2 > (\sigma_2)^2 := \frac{8J_1^4|E(\mathbb{G})|\, M_{max}^4}{(1-\gamma_1)(1-\gamma_2)J_3^6\lambda_2^3}$, condition (11.38) can only be satisfied if $d_{max}^2 = 0 \Leftrightarrow Q_k = Q_j\ \forall k, j$. □

Bibliography

[1] P.A. Absil, R. Mahony, and R. Sepulchre. Riemannian geometry of Grassmann manifolds with a view on algorithmic computation. *Acta Appl. Math.*, 80(2):199–220, 2004.

[2] P.A. Absil, R. Mahony, and R. Sepulchre. *Optimization Algorithms on Matrix Manifolds*. Princeton University Press, 2007.

[3] D. Aeyels and F. De Smet. A model for the dynamical process of cluster formation. *Proc. IFAC NOLCOS Symp.*, 2007.

[4] G. Antonelli and S. Chiaverini. Kinematic control of platoons of autonomous vehicles. *IEEE Trans. Robotics*, 22(6):1285–1292, 2006.

[5] V. Arnold. *Mathematical methods of classical mechanics*. Springer, 1989.

[6] V. Arsigny, X. Pennec, and N. Ayache. Bi-invariant means in Lie groups. Application to left-invariant polyaffine transformations. *INRIA research report*, 5885, 2006.

[7] A. Barg. Extremal problems of coding theory. In H. Niederreiter, editor, *Coding Theory and Cryptography*, Notes of lectures at the Inst. Math. Sci. of Singapore Univ., pages 1–48. World Scientific, 2002.

[8] A. Barg and D.Y. Nogin. Bounds on packings of spheres in the Grassmann manifold. *IEEE Trans. Information Theory*, 48(9):2450–2454, 2002.

[9] C. Beugnon, E. Buvat, M. Kersten, and S. Boulade. GNC design for the DARWIN spaceborne interferometer. *Proc. 6th ESA Conf. Guidance, Navigation and Control Systems*, 2005.

[10] A.M. Bloch, D.E. Chang, N.E. Leonard, and J.E. Marsden. Controlled Lagrangians and the stabilization of mechanical systems II: potential shaping. *IEEE Trans. Automatic Control*, 46(10):1556–1571, 2001.

[11] A.M. Bloch, P.S. Krishnaprasad, J.E. Marsden, and G. Sánchez. Stabilization of rigid body dynamics by internal and external torques. *Automatica*, 28:745–756, 1992.

[12] A.M. Bloch, N.E. Leonard, and J.E. Marsden. Controlled Lagrangians and the stabilization of mechanical systems I: the first matching theorem. *IEEE Trans. Automatic Control*, 45(12):2253–2270, 2000.

[13] V.D. Blondel, J.M. Hendrickx, A. Olshevsky, and J.N. Tsitsiklis. Convergence in multiagent coordination, consensus and flocking. *Proc. 44th IEEE Conf. Decision and Control*, 2005.

[14] V.D. Blondel, J.M. Hendrickx, and J.N. Tsitsiklis. On Krause's consensus formation model with state-dependent connectivity. *to be published in IEEE Trans. Automatic Control*, arxiv 0807.2028v1, 2008.

[15] A.K. Bondhus, K.Y. Pettersen, and J.T. Gravdahl. Leader/follower synchronization of satellite attitude without angular velocity measurements. *Proc. 44th IEEE Conf. Decision and Control*, pages 7270–7277, 2005.

[16] S. Bonnabel, Ph. Martin, and P. Rouchon. Symmetry-preserving observers. *IEEE Trans. Automatic Control*, 53(11):2514–2526, 2008.

[17] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah. Randomized gossip algorithms. *IEEE Trans. Information Theory (Special issue)*, 52(6):2508–2530, 2006.

[18] R.W. Brockett. Dynamical systems that sort lists, diagonalize matrices, and solve linear programming problems. *Lin. Alg. Appl.*, 146:79–91, 1991.

[19] R.W. Brockett. Differential geometry and the design of gradient algorithms. *Proc. Symp. Pure Math., AMS*, 54(1):69–92, 1993.

[20] M. Brookes. Matrix reference manual. *Imperial College London*, 1998-2005.

[21] F. Bullo. Exponential stabilization of relative equilibria for mechanical systems with symmetries. *Proc. 13th Intern. Symp. Math. Theory of Networks and Systems*, 1998.

[22] F. Bullo and R. Murray (advisor). Nonlinear control of mechanical systems: a Riemannian geometry approach. *PhD Thesis, CalTech*, 1998.

[23] S.R. Buss and J.P. Fillmore. Spherical averages and applications to spherical splines and interpolation. *ACM Trans. Graphics*, 20(2):95–126, 2001.

[24] E. Canale and P. Monzón. Gluing Kuramoto coupled oscillator networks. *Proc. 46th IEEE Conf. Decision and Control*, pages 4596–4601, 2007.

[25] M.P. Do Carmo. *Differential Geometry of Curves and Surfaces*. Prentice Hall, 1976.

[26] M.P. Do Carmo. *Riemannian Geometry*. Birkhaeuser, 1992.

[27] F.R.K. Chung. *Spectral Graph Theory*. Number 92 in Regional Conference Series in Mathematics. AMS, 1997.

[28] J.H. Conway, R.H. Hardin, and N.J.A. Sloane. Packing lines, planes, etc.: packings in Grassmannian spaces. *Exper.Math.*, 5(2):139–159, 1996.

[29] J. Cortés, S. Martínez, and F. Bullo. Coordinated deployment of mobile sensing networks with limited-range interactions. *Proc. 43rd IEEE Conf. Decision and Control*, pages 1944–1949, 2004.

[30] J. Cortés, S. Martínez, and F. Bullo. Robust rendez-vous for mobile autonomous agents via proximity graphs in arbitrary dimensions. *IEEE Trans. Automatic Control*, 51(8):1289–1298, 2006.

[31] I.D. Couzin, J. Krause, R. James, G.D. Ruxton, and N.R. Franks. Collective memory and spatial sorting in animal groups. *J. Theor. Biology*, 218:1–11, 2002.

[32] F. Cucker and S. Smale. Emergent behavior in flocks. *IEEE Trans. Automatic Control*, 52(5):852–862, 2007.

[33] F.X. Dechaume-Moncharmont, A. Dornhaus, A.I. Houston, J.M. McNamara, J.E. Collins, and N.R. Franks. The hidden cost of information in collective foraging. *Proc. Biol. Sci.*, 272(1573):1689–1695, 2005.

[34] E. Delhez and J.C.J. Nihoul. *Mécanique rationnelle: modèle mathématique de Newton.* E. Riga Editor, 1996.

[35] J.P. Desai, J.P. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Trans. Robotics and Automation*, 17(6):905–908, 2001.

[36] R. Diestel. *Graph Theory.* Springer, 1997.

[37] T.D. Downs. Orientation statistics. *Biometrika*, 59:665–676, 1972.

[38] A. Edelman, T.A. Arias, and S.T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl.*, 20(2):303–353, 1999.

[39] M.G. Hinchey et al. Swarm-based space exploration. *ERCIM News*, (64):26–27, 2006.

[40] Canadian Air Force. Pilot's manual for basic flying training. *TC-44*, 1962.

[41] G.A. Galperin. A concept of the mass center of a system of material points in the constant curvature spaces. *Comm. Math. Phys.*, 154:63–84, 1993.

[42] E. Goles-Chacc, F. Fogelman-Soulie, and D. Pellegrin. Decreasing energy functions as a tool for studying threshold networks. *Discr. Appl. Math.*, 12(3):261–277, 1985.

[43] S. Goss, S. Aron, J.L. Deneubourg, and J.M. Pasteels. Self-organized shortcuts in the argentine ant. *Naturwissenschaften*, 76:579–581, 1989.

[44] D. Groisser. Newton's method, zeroes of vector fields, and the Riemannian center of mass. *Adv. Appl. Math.*, 33:95–135, 2004.

[45] P. Gruber and F.J. Theis. Grassmann clustering. *Proc. 14th Eur. Signal Processing Conf.*, 2006.

[46] C.D. Hall, P. Tsiotras, and H. Shen. Tracking rigid body motion using thrusters and momentum wheels. *J. Astronautical Sci.*, 50(3):311–323, 2002.

[47] W.R. Hamilton. Lectures on quaternions. *Royal Irish Academy*, 1853.

[48] H. Hanssmann, N.E. Leonard, and T.R. Smith. Symmetry and reduction for coordinated rigid bodies. *Eur. J. Control*, 12(2):176–194, 2006.

[49] U. Helmke and J.B. Moore. *Optimization and Dynamical Systems.* Springer, 1994.

[50] J. Hendrickx and V. Blondel (advisor). Graphs and networks for the analysis of autonomous agent systems. *PhD Thesis, Université Catholique de Louvain*, 2008.

[51] J.J. Hopfield. Neural networks and physical systems with emergent collective computational capabilities. *Proc. Nat. Acad. Sci.*, 79:2554–2558, 1982.

[52] K. Hueper and J. Manton. The Karcher mean of points on $SO(n)$. *Talk at Cesame (UCL, Belgium)*, 2004.

[53] S.P. Hughes and C.D. Hall. Optimal configurations of rotating spacecraft formations. *J. Astronautical Sci.*, 48(2-3):225–247, 2000.

[54] M. Hurley. Chain recurrence, semiflows, and gradients. *J. Dynam. Diff. Eq.*, 7(3):437–456, 1995.

[55] D. Izzo and L. Pettazzi. Equilibrium shaping: distributed motion planning for satellite swarm. *Proc. 8th Intern. Symp. Artificial Intelligence, Robotics and Automation in Space*, 2005.

[56] A. Jadbabaie, J. Lin, and A.S. Morse. Coordination of groups of mobile autonomous agents using nearest neighbor rules. *IEEE Trans. Automatic Control*, 48(6):988–1001, 2003.

[57] A. Jadbabaie, N. Motee, and M. Barahona. On the stability of the Kuramoto model of coupled nonlinear oscillators. *Proc. Amer. Control Conf.*, 2004.

[58] S.M. Jalnapurkar and J.E. Marsden. Stabilization of relative equilibria II. *Regul.Chaotic Dyn.*, 3(3):161–179, 1998.

[59] J. Jeanne, D. Paley, and N.E. Leonard. On the stable phase relationships of loop-coupled planar particles. *Proc. 44th IEEE Conf. Decision and Control*, 2005.

[60] V. Jurdjevic. *Geometric Control Theory*. Cambridge University Press, 1997.

[61] V. Jurdjevic and J.P. Quinn. Controllability and stability. *J. Diff. Eq.*, 28(3):381–389, 1978.

[62] E. Justh and P. Krishnaprasad. A simple control law for UAV formation flying. Technical report, TR 2002-38, ISR, University of Maryland, 2002.

[63] E.W. Justh and P.S. Krishnaprasad. Equilibria and steering laws for planar formations. *Systems and Control Letters*, 52:25–38, 2004.

[64] E.W. Justh and P.S. Krishnaprasad. Natural frames and interacting particles in three dimensions. *Proc. 44th IEEE Conf. Decision and Control*, pages 2841–2846, 2005.

[65] H. Karcher. Riemannian center of mass and mollifier smoothing. *Comm. Pure and Appl. Math.*, 30:509–541, 1977.

[66] H.K. Khalil. *Nonlinear Systems*. Prentice Hall, 2nd edition edition, 1996.

[67] O. Khatib. Real-time obstacle avoidance for manipulators and mobile robots. *Int. J. Robotics Research*, pages 90–99, 1986.

[68] K.A. Krakowski and L. Noakes (advisor). Geometrical methods of inference. *PhD Thesis, University of Western Australia*, 2002.

[69] U. Krause. Soziale Dynamiken mit vielen Interakteuren. Eine Problemskizze. *Modellierung und Simulation von Dynamiken mit vielen interagierenden Akteuren*, pages 37–51, 1997.

[70] T.R. Krogstad and J.T. Gravdahl. Coordinated attitude control of satellites in formation. In *Group Coordination and Cooperative Control*, volume 336 of *Lect. N. Control and Information Sci.*, chapter 9, pages 153–170. Springer, 2006.

[71] Y. Kuramoto. Self-entrainment of population of coupled nonlinear oscillators. In *Internat. Symp. on Math. Problems in Theoretical Physics*, volume 39 of *Lect. N. Physics*, page 420. Springer, 1975.

[72] J.R. Lawton and R.W. Beard. Synchronized multiple spacecraft rotations. *Automatica*, 38:1359–1364, 2002.

[73] D. Lee and M.W. Spong. Stable flocking of multiple inertial agents on balanced graphs. *IEEE Trans. Automatic Control*, 52(8):1469–1475, 2007.

[74] N.E. Leonard. Stabilization of underwater vehicle dynamics with symmetry-breaking potentials. *Systems and Control Letters*, 32:35–42, 1997.

[75] N.E. Leonard and P.S. Krishnaprasad (advisor). Averaging and motion control of systems on Lie groups. *PhD Thesis, University of Maryland, College Park*, 1994.

[76] N.E. Leonard, D. Paley, F. Lekien, R. Sepulchre, D. Frantantoni, and R. Davis. Collective motion, sensor networks and ocean sampling. *Proc. IEEE*, 95(1):48–74, 2007.

[77] J. Lin, A.S. Morse, and B.D.O. Anderson. The multi-agent rendez-vous problem (Part 1: the synchronous case; Part 2: the asynchronous case). *SIAM J. Control and Optimization*, 46(6):2096–2147, 2007.

[78] Z.X. Liu and L. Guo. Connectivity and synchronization of Vicsek model. *Science in China Series F: Information Sciences*, 51(7):848–858, 2008.

[79] A. Machado and I. Salavessa. Grassmannian manifolds as subsets of Euclidean spaces. *Res. Notes in Math.*, 131:85–102, 1985.

[80] D.H.S. Maithripala, J.M. Berg, and W.P. Dayawansa. Almost-global tracking of simple mechanical systems on a general class of Lie groups. *IEEE Trans. Automatic Control*, 51(1):216–225, 2006.

[81] J.E. Marsden. *Lectures on Mechanics*. Cambridge University Press, 1992.

[82] J.E. Marsden, T. Ratiu, and A. Weinstein. Semidirect products and reduction in mechanics. *AMS Trans.*, 281(1):147–177, 1984.

[83] J.E. Marsden and T.S. Ratiu. *Introduction to Mechanics and Symmetry*. Springer, 1994.

[84] N. C. Martins. Optimal design of formations for wireless networks of mobile and static agents. *Workshop "Cooperative Control of Multiple Autonomous Vehicles" pre-IFAC World Congress*, 2008.

[85] C.R. McInnes. Potential function methods for autonomous spacecraft guidance and control. *Proc. AAS/AIAA Astrodynamical Specialists Conf.*, (95-447), 1995.

[86] C.R. McInnes. Distributed control for on-orbit assembly. *Adv. Astronautical Sci.*, 90:2079–2092, 1996.

[87] M. Mesbahi. State-dependent graphs. *Proc. 42nd IEEE Conf. Decision and Control*, pages 3058–3063, 2003.

[88] M. Mesbahi. On state-dependent dynamic graphs and their controllability properties. *IEEE Trans. Automatic Control*, 50(3):387–392, 2005.

[89] M. Mesbahi and F.Y. Hadaegh. Formation flying control of multiple spacecraft via graphs, matrix inequalities, and switching. *J. Guidance, Control and Dynamics*, 24(2):369–377, 2001.

[90] J. Milnor. Differential topology. volume 2 of *Lectures on Modern Mathematics*, pages 165–183. Wiley, 1964.

[91] K. Mischaikow, H. Smith, and H.R. Thieme. Asymptotically autonomous semi-flows, chain recurrence and Lyapunov functions. *AMS Trans.*, 347(5):1669–1685, 1995.

[92] M. Moakher. Means and averaging in the group of rotations. *SIAM J. Matrix Anal. Appl.*, 24(1):1–16, 2002.

[93] J.D. Monnier. Optical interferometry in astronomy. *Reports on Progress in Physics*, 66:789–857, 2003.

[94] L. Moreau. Stability of continuous-time distributed consensus algorithms. *Proc. 43rd IEEE Conf. Decision and Control*, pages 3998–4003, 2004.

[95] L. Moreau. Stability of multi-agent systems with time-dependent communication links. *IEEE Trans. Automatic Control*, 50(2):169–182, 2005.

[96] S. Nair and N.E. Leonard (advisor). Stabilization and synchronization of networked mechanical systems. *PhD Thesis, Princeton University*, 2006.

[97] S. Nair and N.E. Leonard. Stabilization of a coordinated network of rotating rigid bodies. *Proc. 43rd IEEE Conf. Decision and Control*, pages 4690–4695, 2004.

[98] S. Nair and N.E. Leonard. Stable synchronization of rigid body networks. *Networks and Heterogeneous Media*, 2(4):595–624, 2007.

[99] S. Nair and N.E. Leonard. Stable synchronization of mechanical system networks. *SIAM J. Control and Optimization*, 47(2):661–683, 2008.

[100] S. Nair, N.E. Leonard, and L. Moreau. Coordinated control of networked mechanical systems with unstable dynamics. *Proc. 42nd IEEE Conf. Decision and Control*, pages 550–555, 2003.

[101] R. Olfati-Saber. Flocking for multi-agent dynamic systems: algorithms and theory. *IEEE Trans. Automatic Control*, 51(3):401–420, 2006.

[102] R. Olfati-Saber, J.A. Fax, and R.M. Murray. Consensus and cooperation in networked multi-agent systems. *Proc. IEEE*, 95(1):215–233, 2007.

[103] R. Olfati-Saber and R.M. Murray. Graph rigidity and distributed formation stabilization of multi-vehicle systems. *Proc. 41st IEEE Conf. Decision and Control*, 2002.

[104] R. Olfati-Saber and R.M. Murray. Consensus problems in networks of agents with switching topology and time delays. *IEEE Trans. Automatic Control*, 49(9):1520–1533, 2004.

[105] A. Olshevsky and J.N. Tsitsiklis. Convergence rates in distributed consensus and averaging. *Proc. 45th IEEE Conf. Decision and Control*, pages 3387–3392, 2006.

[106] A.L. Onishchik, V.V. Gorbatsevich, and E. Borisovich Vinberg. *Lie groups and Lie algebras I (translation of Russian original)*. Birkhaeuser, 1993.

[107] P. Oprocha. Topological approach to chain recurrence in continuous dynamical systems. *Opuscula Mathematica*, 25(2):261–268, 2005.

[108] X. Pennec. Probabilities and statistics on Riemannian manifolds: a geometric approach. *INRIA research report*, 5093, 2004.

[109] Wei Ren. Distributed attitude consensus among multiple networked spacecraft. *Proc. American Control Conference*, 2006.

[110] Wei Ren. Distributed attitude alignment in spacecraft formation flying. *Int. J. of Adaptive Control and Signal Processing*, 21(2–3):95–113, 2007.

[111] Wei Ren. Multi-vehicle consensus with a time-varying reference state. *Systems and Control Letters*, 56:474–483, 2007.

[112] Wei Ren and R.W. Beard. *Distributed Consensus in Multi-vehicle Cooperative Control*. Communications and Control Engineering. Springer, 2008.

[113] C.W. Reynolds. Flocks, herds, and schools: a distributed behavioral model. *Comp. Graph. (ACM SIGGRAPH Conf. Proc.)*, 21:25–34, 1987.

[114] E. Rimon and D.E. Koditschek. Exact robot navigation using artificial potential functions. *IEEE Trans. Robotics and Automation*, pages 501–518, 1992.

[115] A. Sarlette, S. Bonnabel, and R. Sepulchre. Coordinated motion design on Lie groups. *submitted to IEEE Trans. Automatic Control*, arxiv 0807.4416, 2008.

[116] A. Sarlette, S. Bonnabel, and R. Sepulchre. Coordination on Lie groups. *Proc. 47th IEEE Conf. Decision and Control*, pages 1275–1279, 2008.

[117] A. Sarlette and R. Sepulchre. Consensus optimization on manifolds. *to be published in SIAM J. Control and Optimization*, 2008.

[118] A. Sarlette, R. Sepulchre, and N.E. Leonard. Discrete-time synchronization on the N-torus. *Proc. 17th Intern. Symp. Math. Theory of Networks and Systems*, pages 2408–2414, 2006.

[119] A. Sarlette, R. Sepulchre, and N.E. Leonard. Autonomous rigid body attitude synchronization. *Proc. 46th IEEE Conf. Decision and Control*, pages 2566–2571, 2007.

[120] A. Sarlette, R. Sepulchre, and N.E. Leonard. Cooperative attitude synchronization in satellite swarms: a consensus approach. *Proc. 17th IFAC Symp. Automatic Control in Aerospace*, 2007.

[121] A. Sarlette, R. Sepulchre, and N.E. Leonard. Autonomous rigid body attitude synchronization. *to be published in Automatica*, 2008.

[122] A. Sarlette, S.E. Tuna, V.D. Blondel, and R. Sepulchre. Global synchronization on the circle. *Proc. 17th IFAC World Congress*, 2008.

[123] L. Scardovi, N.E. Leonard, and R. Sepulchre. Stabilization of three-dimensional collective motion. *submitted to the Brockett Legacy Special Issue of Communications in Information and Systems*, 2008.

[124] L. Scardovi, A. Sarlette, and R. Sepulchre. Synchronization and balancing on the N-torus. *Systems and Control Letters*, 56(5):335–341, 2007.

[125] L. Scardovi and R. Sepulchre. Synchronization in networks of identical linear systems. *Proc. 47th IEEE Conf. Decision and Control*, pages 546–551, 2008.

[126] L. Scardovi, R. Sepulchre, and N.E. Leonard. Stabilization of collective motion in three dimensions: a consensus approach. *Proc. 46th IEEE Conf. Decision and Control*, pages 2931–2936, 2007.

[127] B.F. Schutz. *A first course in general relativity.* Cambridge University Press, 1984.

[128] R. Sepulchre, M. Janković, and P. Kokotović. *Constructive Nonlinear Control.* Springer, 1997.

[129] R. Sepulchre, D. Paley, and N.E. Leonard. Collective motion and oscillator synchronization. In A. Morse V. Kumar, N. Leonard, editor, *Cooperative control*, volume 309 of *Lect. N. Control and Information Sci.*, pages 189–205. Springer, 2004.

[130] R. Sepulchre, D. Paley, and N.E. Leonard. Group coordination and cooperative control of steered particles in the plane. In *Group Coordination and Cooperative Control*, volume 336 of *Lect. N. Control and Information Sci.*, chapter 13, pages 217–232. Springer, 2006.

[131] R. Sepulchre, D. Paley, and N.E. Leonard. Stabilization of planar collective motion with all-to-all communication. *IEEE Trans. Automatic Control*, 52(5):811–824, 2007.

[132] R. Sepulchre, D. Paley, and N.E. Leonard. Stabilization of planar collective motion with limited communication. *IEEE Trans. Automatic Control*, 53(3):706–719, 2008.

[133] J.C. Simo, D.R. Lewis, and J.E. Marsden. Stability of relative equilibria I: the reduced energy momentum method. *Arch. Rat. Mech. Anal.*, 115:15–59, 1991.

[134] S. Singh and W. Yim. Nonlinear adaptive backstepping design for spacecraft attitude control using solar radiation pressure. *Proc. 41st IEEE Conf. Decision and Control*, pages 1239–1244, 2002.

[135] T.R. Smith, H. Hanssmann, and N.E. Leonard. Orientation control of multiple underwater vehicles with symmetry-breaking potentials. *Proc. 40th IEEE Conf. Decision and Control*, pages 4598–4603, 2001.

[136] E. D. Sontag. Remarks on stabilization and input-to-state stability. *Proc. 28th IEEE Conf. Decision and Control*, pages 1376–1378, 1989.

[137] S.H. Strogatz. From Kuramoto to Crawford: exploring the onset of synchronization in populations of coupled nonlinear oscillators. *Physica D*, 143:1–20, 2000.

[138] S.H. Strogatz. *Sync: The emerging science of spontaneous order*. Hyperion, 2003.

[139] H. Struemper and P.S. Krishnaprasad (advisor). Motion control for nonholonomic systems on matrix Lie groups. *PhD Thesis, University of Maryland*, 1998.

[140] J. Stuelpnagel. On the parametrization of the three-dimensional rotation group. *SIAM Rev.*, 6(4):422–430, 1964.

[141] D.T. Swain, N.E. Leonard, I.D. Couzin, A. Kao, and R. Sepulchre. Alternating spatial patterns for coordinated group motion. *Proc. 46th IEEE Conf. Decision and Control*, pages 2925–2930, 2007.

[142] D. Swaroop and J.K. Hedrick (advisor). String stability of interconnected systems: an application to platooning in automated highway systems. *PhD Thesis, University of California Berkeley*, 1994.

[143] J.N. Tsitsiklis and M. Athans (advisor). Problems in decentralized decision making and computation. *PhD Thesis, MIT*, 1984.

[144] J.N. Tsitsiklis and D.P. Bertsekas. Distributed asynchronous optimal routing in data networks. *IEEE Trans. Automatic Control*, 31(4):325–332, 1986.

[145] J.N. Tsitsiklis, D.P. Bertsekas, and M. Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Trans. Automatic Control*, 31(9):803–812, 1986.

[146] A.J. van der Schaft. Stabilization of Hamiltonian systems. *Nonlinear Analysis, Theory, Methods and Applications*, 10:1021–1035, 1986.

[147] M.C. VanDyke and C.D. Hall. Decentralized coordinated attitude control of a formation of spacecraft. *J. Guidance, Control and Dynamics*, 29(5):1101–1109, 2006.

[148] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet. Novel type of phase transition in a system of self-driven particles. *Phys. Rev. Lett.*, 75(6):1226–1229, 1995.

[149] J. C. Willems. Lyapunov functions for diagonally dominant systems. *Automatica*, 12:519–523, 1976.

[150] M.K.S. Yeung and S.H. Strogatz. Time delay in the Kuramoto model of coupled oscillators. *Physical Review Letters*, 82(3):648–651, 1999.