Flag Coordination Games

Kohan Marzagao, David

Awarding institution:
King's College London

Flag Coordination Games

David Kohan Marzagão

A thesis submitted in partial fulfilment for
the degree of Doctor of Philosophy

in the
Department of Informatics
School of Natural & Mathematical Sciences
King's College London

2019

To my parents, Sarah and Laercio,
and to the memory of my grandmother Bertha,
for their unconditional support.

Abstract

Many multi-agent coordination problems can be understood as a sequence of autonomous local choices between a finite set of options, with each local choice undertaken simultaneously without explicit coordination between decision-makers, and with a shared goal of achieving a desired global state or states. Examples of such problems include classic consensus problems between nodes in a distributed computer network and the adoption of competing technology standards. In this thesis, we model such problems as a multi-round game between agents having flags of different colours to represent the finite choice options, and all agents seeking to achieve global patterns of colours through a succession of local colour-selection choices.

We generalise and formalise the problem, proving results for the probabilities of achievement of common desired global states when these games are undertaken on directed or undirected bipartite graphs, extending known results that require graphs to be non-bipartite. We also calculate probabilities for the game entering infinite cycles of non-convergence. In addition, we present a game-theoretic approach to the problem that has a mixed-strategy Nash equilibrium where two players can simultaneously flip the colour of one of the opponent's nodes in an arbitrary directed graph before or during a Flag Coordination Game.

A known hard problem in consensus protocols consists of the introduction of a bias towards a given opinion. Such problems in a general graph are unlikely to have an analytic solution, however, for cycles we provide the probabilities of convergence for each colour based on the initial configuration of the game.

We apply results on Flag Coordination Games into the Theory of Argumentation. We consider two teams of agents engaging in a debate to persuade an audience of the acceptability of a central argument. This is

modelled by a bipartite abstract argumentation framework with a distinguished topic argument, where each argument is asserted by a distinct agent. One partition defends the topic argument and the other partition attacks the topic argument. The dynamics are based on Flag Coordination Games: in each round, each agent decides whether to assert its argument based on local knowledge. The audience can see the induced sub-framework of all asserted arguments in a given round, and thus the audience can determine whether the topic argument is acceptable, and therefore which team is winning. We derive an analytical expression for the probability of either team winning given the initially asserted arguments, where in each round, each agent probabilistically decides whether to assert or withdraw its argument given the number of attackers.

Acknowledgements

‘First and foremost’ is not enough to convey how *sine qua non* Peter McBurney’s supervision was to my doctoral studies. I wholeheartedly thank him for his guidance, support, kindness and encouragement. His brilliance will always inspire me.

I am especially grateful to my second supervisor, Kathleen Steinhöfel for not letting me accommodate and to always encourage me to improve my work. Her knowledge and expertise in both and theoretical and applied methods were key in helping me refine my research.

Thanks also to Paul Dunne and Nicolas Maudet for kindly agreeing to take the time to examine this thesis, and for their detailed comments on the work.

I want to thank all my coauthors, with whom I learned immensely. Josh Murphy, Antony Peter Young, Colin Cooper, Marcelo Matheus Gauy, Nicolas Rivera, Elizabeth Black, and Michel Luck. Working with Peter Young showed me that the number of theorems one can prove between one bus stop and the next is greater than zero. I want to thank once again my friend and coauthor Josh, whose enthusiasm, intelligence, and Britishness, have, respectively, driven, inspired, and amused me during these past years.

Teaching was an integral part of my studies, and I am deeply grateful to all the academics that I had the privilege of working with. Amanda Coles, Maxime Crochemore, Luca Viganò, Sameer Murthy, Hana Chockler, Simon Solomon, Kathleen Steinhofel, Brendan Michael, and Ioannis Kouletsis. Luca showed me how Cryptography can be taught in such an engaging way. I want to thank Ioannis for giving me the pleasure of working with him in his beautifully crafted methods course. Finally, I am especially grateful to Dan Abramson, whose enormous enthusiasm and confidence let a first-year PhD student run a school-wide mathematics challenge.

Finally, I thank CNPq for fully funding my studies.

From this point onward, I will be taking the possibly inappropriate path of thanking people that were not directly related my PhD work, but have been part of my life in these long PhD years, and without their support things would be way more difficult.

I thank King's College London, especially the Informatics Department and all the PhD students that come with it. Coffee breaks within coffee breaks might have made short term deadlines more challenging, but certainly enhanced research it in the long run. Samhar Mahmoud and Lina Barakat, for showing me the art of Trex. Christos Hadjinikolis, for making me feel at home in the very early days of my PhD. Martin Chapman, Chris Hampson, Wiktor Piotrowski, Emre Savas, Josef Bajada, Jonathan Cardoso, and Tomas Vitek, for making the 6th floor Strand a great place go to everyday. Tanja Daub, for being such an amazing companion in this journey. Bush House would not be the same without the strangest of groups: Yani Moraru, Stefan Sarkadi, Francesca Mosca, Diego Sempreboni, Panos Charalampopoulos, Angelos Gkikas, Andreas Xydis, Federico Castagna, Jake Swambo, Parisa Zehtabi, Zhouling Zhang, Leonardo Cunha, and many others.

Goodenough College has been my home for most of this journey. It is hard to try to look back and imagine not having met all these wonderful people in such a unique place. Vasilios Mavroudis, for the late-night cooking, and lifelong discussions. Malu Gatto and Hylo Gurgel, for the sharing and discovery. Zak Rosentzveig and Gabrielle Jacobs, for the coffee and the music. Thiago Oliveira and Louise Giansante, for having put up with me on a daily basis. Wilhelm Rosenberg, Mary Walker, Sarah Speziali, Ariana Hübner, Kate McNeil, Roberta Alessandrini, Beatriz Canseco, Luisa Olander, Valeria Valotto, Rani Suleman, Emmy Stavropoulou, and so many others for their time and effort to make our college even better. Finally, I thank Luisa Sette, Damian Sercombe, Thea Christophersen, Aleksandra Ziemiszewska, Ciro Moraes dos Reis, Marina Bezzi, Jardiel Nogueira, Larissa Boratti, Marcelo Ilarraz, Tom Pugh, Marina Avelar, Igor Maia, Lucía Sánchez, Eva Blanco, Nora Ratzmann, and Filippo Temporin; the college would not have been the same without them.

My dear musician friends, especially those who make *Regional do Grafton* the best place to be on a Monday evening. Among all, Giovanna Carloni and Julia Garcia have most consistently endured my invoking of this thesis as the reason to miss practices.

I want to deeply thank Deborah Raphael for being the reason I have chosen King's College London in the first place, and for the encouragement and support throughout my undergraduate studies. I also extend my gratitude to Roberto Moisés and Mônica Torkomian, because not rarely early mathematical adventures bear fruits sometime in a distant future.

I am grateful to these amazing people from all around that somehow did not fit in the previous paragraphs. Daniel Bittencourt, Francisco Viríssimo, Roberto Baldijão, Luisa Buzzo, Julia Franco, Sofia Fichino, Clara Kok Martins, Miguel Fausto, Luiz Felipe Orlando, Anita Pompéia, Lucas Machado, Gilson Reis, Tiago Madeira, Mônica Galvão, Adrian Fuentes, Arthur Rachman, Aleksandra Aloric, Pablo De Castro, Beatriz Zilberman, Ana Paula Lopes, Maria Sette, Alexandre Angulo, Beatriz Sakashita, Giulia Afiune, Jonathan Bootle, Alexandre Denigres, Lucia Furlan, Alice Mahlmeister, Vitor Lopes, Lázaro Assunção, Bartira Maués, Lais Kohan, and André Kohan.

I thank my parents, Sarah Kohan and Laercio Marzagão, for the unconditional support to my studies, since far before I knew what a graph was.

Without Luciana Basualdo Bonatto, this thesis would not be a reality. Any attempt to convey how fortunate I am to walk alongside her would be an understatement.

Contents

Chapter 1

Introduction

1.1 Motivation

Many multi-agent coordination problems may be represented as a collection of agents choosing autonomously from a finite set of options using only limited information, while sharing a common desire for a global state. For example, users of a new technology choosing between alternative technical standards each face the same choice of possible options, but make their choices without necessarily knowing the choices of others. In the case of *network goods* [69], the utilities of each option to any one user depend on the choices made by the other users; in the classic example, a fax machine is only of value to any one company if the organisations with which that company communicates also have fax machines. Hence, potential adopters may choose the option they believe most others will choose [77]. Even for non-technology products, such as clothes and food, consumers might gain additional benefits from purchasing products or services that they believe have been chosen (or not chosen) by other consumers, over any perceived benefits of the good or service itself.

In these cases, agents might wish to all adopt the same choice as one another, so that the desired shared global state is one of *consensus*. In other cases, the global state might have a different pattern, for example, a sequence of alternating states. For instance, in a robot bucket brigade, each robot in a line would need to be either in a giving state or in a receiving state at each time step, and in the complementary state to each of its neighbours at that time step. At each subsequent time step, each robot would need to switch to the other state.

We can model such situations as an abstract multi-agent game of flag colouring, where the different flag colours represent the different decision options each agent faces. While there are applications where the desired global state of the system needs to be achieved in a single step [46], we consider only cases where the agents proceed in a sequence of rounds, making individual choices simultaneously at each step. If at any step, a desired global state is achieved, the game ends. Otherwise, it continues.

As a motivating example, consider the context of robot fire brigades, in which robots are expected to be able to replace firefighters by performing rescue missions in buildings on fire. We do not expect a human to accompany robots, therefore human orders cannot be given regarding the best way to conduct this operation (e.g., which room should each robot go to and when). Furthermore, there might be no time or means for a conversation to take place between the AIs, and therefore robots might have to decide what to do on the basis of only the action of the other robots and their local environment.

With that motivation in mind, we define a Flag Coordination Game as a framework to study distributed processes, with no central authority involved, and in which the only information each agent can broadcast is their current state. In broad terms, Flag Coordination Games encompass both consensus games on graphs, in which each node copies a neighbour to seek a global consensus, and distributed proper colouring of graphs, in which nodes want to move away from neighbours' opinions. Flag Coordination Games can describe both randomised processes such as random walks on a graph and deterministic ones such as Conway's Game of Life. Voting protocols, and disease-spreading processes, are further examples of processes that can be seen as Flag Coordination Games. The particularity of such games is that the decentralised decisions only take into account agents' states, with no additional information shared between agents within the network.

There are many possible variations on this general situation. We illustrate some of them before formally defining Flag Coordination Games in the next chapter.

(i) We assume a finite set of autonomous agents, possibly with a shared clock, with each empowered to decide between a finite set of decision options at different points in time. These options may or may not be the same for every agent and decisions may or may not be made synchronously, at successive time steps. For simplicity, the decision options are represented by flags of different colours.

(ii) Agents are connected via a network, and at any given time, each agent is able to see the decisions made by some subset of the set of agents, typically its immediate neighbours, i.e., those agents to which it is directly linked. For generality, we allow the visibility of agents to change throughout this game. Agents do not communicate in any other way with one another.

(iii) Agents know the decision option they themselves choose at each time step but they are not necessarily assumed to have any memory of previous choices, of themselves, of other agents, or of previous global states. Indeed, in this work, we are going to focus on Flag Coordination Games in which agents have no memories.

(iv) Agents all share a desired set of global goal states (possibly just one state) for the collective set of agents. This set of shared global goal states could be, for example, consensus (all agents choose the same decision option) or a global state in which no two connected agents have made the same choice (e.g., alternating flag colours).

(v) We assume that, between one time step and the next, agents are not informed whether or not their previous decisions achieved one of the desired goal states. That is why we will be looking into algorithms under which the global goal states are stable, i.e., a state where the algorithm would not lead agents to change their state. If and when a stable goal state is achieved, we say the sequential decision process ends.

(vi) In most frameworks studied in this thesis, agents are assumed to be well intentioned (i.e., not malicious or whimsical), and bug-free. However, we allow Flag Coordination Game in general to include malicious agents that try to prevent a global goal state to be achieved.

In this thesis, we articulate a formal model (defined as the set of rules of a Flag Coordination Game) for a flag-colouring game, based on these assumptions, with the purpose of answering the following questions:

A1 Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the probability that the sequential decision process will enter an infinite cycle that does not converge to a pre-specified global goal state (i.e., an infinite cycle of non-convergence)?

A2 Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the probability that the sequential decision process will converge to a pre-specified desired global goal state?

A3 Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the expected number of decision rounds (time steps) to reach a pre-specified global goal state?

A4 Which sufficient conditions on the rules of a Flag Coordination Game are such that, for at least one possible initial state, there is a positive probability that the state loop described in Question **A1** is entered?

A5 How can we apply the concept of Flag Coordination Games to the field of Argumentation Theory to study a form of distributed argumentation in which each argument is controlled by an independent agent?

A6 How can a Flag Coordination Game be influenced by external agents?

A7 What is the impact of the introduction of bias towards a given opinion (or flag colour) in the set of rules of a Flag Coordination Game?

A8 Can every state in a Flag Coordination Game be reached from any other state with positive probability?

1.2 Thesis Structure and Contribution

In this section we provide an overview of each chapter of the thesis by summarising its main contributions based on the questions set earlier in this introduction.

Chapter 2 formally defines the set of rules of a Flag Coordination Game, and introduces the necessary technical background and provides a summary of the relevant related work.

In Chapter 3, we focus on the convergence of synchronous consensus protocols. Prior work in this field established probabilities for the convergence-to-consensus for each one of several possible opinions, as well as time bounds for the process to end. All agents change their opinions synchronously taking into account the colours of their neighbours and using a common algorithm. Previous work, however, assumes that the Markov chain that describes this process has only the consensus states as recurrent ones, discarding graphs that might lead to loops in the consensus game.

In the first part of Chapter 3, building on previous work on general graphs by Hassin and Peleg [35], we present results for Questions **A1**, **A2**, and **A3** for bipartite undirected graphs. These are graphs where the nodes can be divided

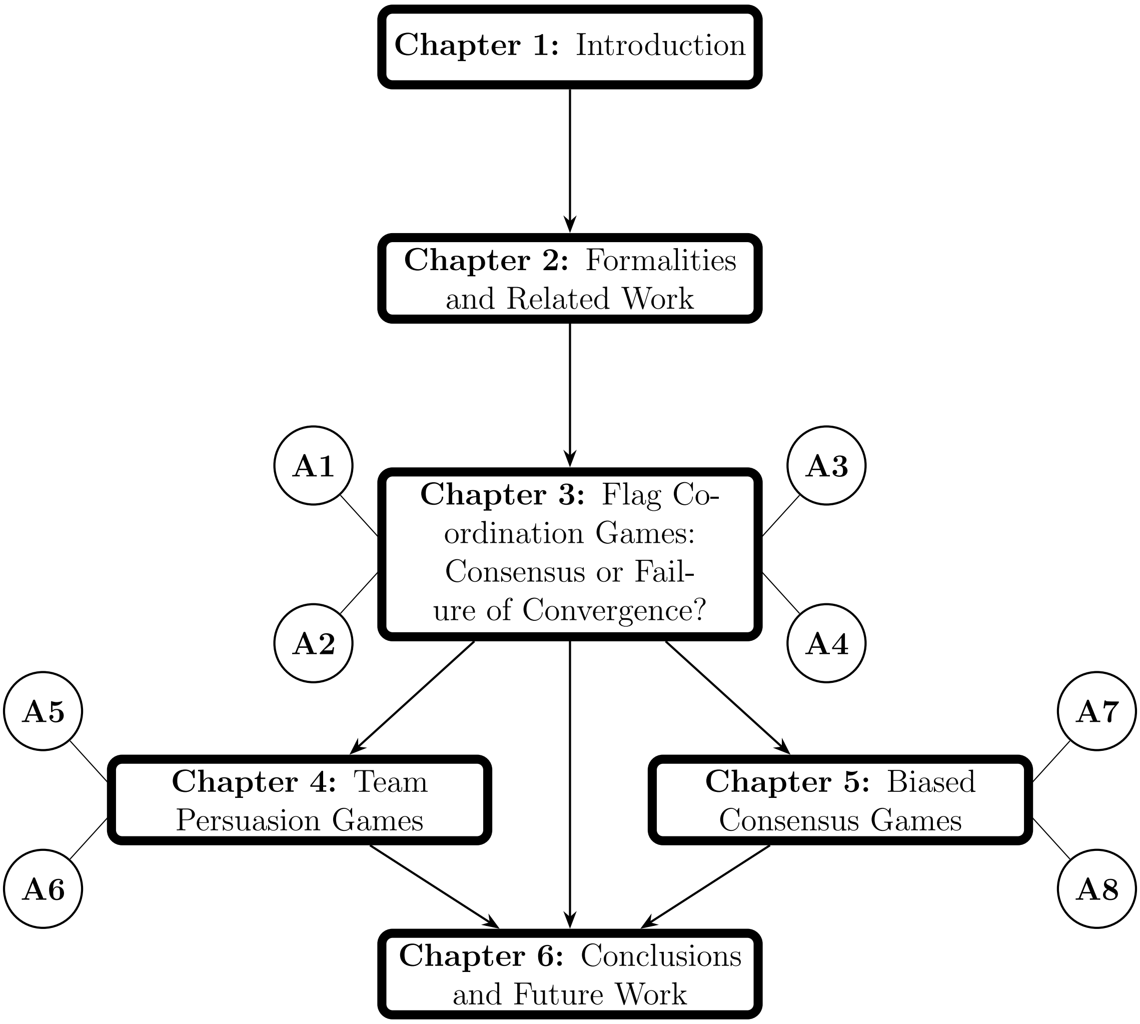


Figure 1.1: Thesis Structure, Chapter Correlation, and Question Index.

naturally into two mutually exclusive types, for example, buyers and sellers in an online marketplace. In the second part of Chapter 3, we fully answer Question **A4** in the domain of consensus games on any directed graph, as well as addressing Questions **A1**, and **A2** for such graphs.

Chapter 4 covers anti-consensus games and computational argumentation theory. We address Question **A5** by introducing a distributed argumentation scenario, in which each node acts as an agent that is an expert in their own knowledge domain. The agent decides whether or not to assert their argument at each round. Our results provide the probabilities that a given argument, the topic, will be accepted or rejected in the long run of this process.

In addition to this analysis and in light of Question **A6**, we consider, also in

Chapter 4, the effect of bribery in such games: we present a game-theoretic approach to the situation which two or more players can simultaneously flip the colour of one of their opponents' nodes in a bipartite graph before or during a flag-coordination game. We also prove that such games, regardless of the number of bribers, always admit a pure strategy Nash equilibrium.

Finally, in order to explore Question **A7**, Chapter 5 introduces a generalisation of the consensus games from Chapter 3 by taking into account processes in which agents have a bias towards a given opinion. Such problems in a general graph are unlikely to have an analytic solution, however, for cycle graphs, a martingale with respect to the Markov chain describing the biased colouring process has been found. This generalisation can be motivated by the following situation: consider a voting process represented by a consensus game on a graph in which a given opinion (or colour) wins. Consider now that a second game takes place in the same graph. It is not unreasonable to expect that the previous result will have an impact on this second process. For example, voters might favour the previously consensual opinion, generating a bias towards this outcome. In the context of biased consensus games, we explore Question **A8** of whether a given state can be reached by another in a cycle graph by introducing a correspondence between such games and a process of self-annihilating random walks.

We present Figure 1.1 to summarise this thesis structure highlighting in which chapter each of these questions are explored. An arrow from Chapter i to Chapter j indicate that results in Chapter j might be based on results from Chapter i or earlier ones. In particular, Chapters 4 and 5 are independent.

1.2.1 A Note on Presentation

Most chapters begin (before the introduction section) with a motivational problem that aims to contextualise what is to be studied subsequently in the chapter. These problems will then be resolved later in the same chapter or, more rarely, in subsequent ones.

Figures in this thesis make extensive use of colours. For that reason, a black-and-white version of every coloured figure was included in Appendix B with reference to the original image and a global key to correspond both versions.

There is often usage of notation-heavy definitions. Because of that, a list of figures, an index, and a list of symbols are added at the end of this thesis.

More lists of questions such as the one above (**A1**, ..., **A8**) will be presented in chapters to follow. To guarantee uniqueness, each new list will be indexed by a different character.

1.3 Publications

This thesis includes results published in the following peer-reviewed papers.

[44] David Kohan Marzagão, Nicolás Rivera, Colin Cooper, Peter McBurney, and Kathleen Steinhöfel. Multi-agent flag coordination games. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 1442–1450. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[43] David Kohan Marzagão, Josh Murphy, Anthony Peter Young, Marcelo M Gauy, Michael Luck, Peter McBurney, and Elizabeth Black. Team Persuasion. In *The 3rd International Workshop on Theory and Applications of Formal Argument (TAFA)*, pages 159–174. Springer, 2017.

The main contribution and results of [44] can be found in Chapter 2 and Chapter 3. Moreover, ideas, such as those related to Question **A6**, were introduced in [44] and developed and further explored in Chapter 4, which also contains the main contribution of [43].

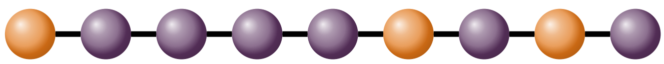
Chapter 2

Formalities and Related Work

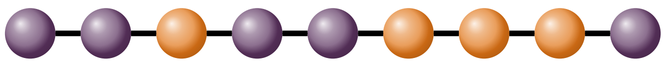
PROBLEM 1 (ROBOT BUCKET BRIGADE). Consider a line formed of autonomous robots that have the shared goal of passing buckets of water in the direction of a building on fire, and empty buckets in the other direction. Each robot has two possible actions: to receive a bucket from each neighbour or to pass a bucket to each neighbour. They want to avoid the situation in which two neighbouring robots are currently taking the same action (neither can both pass to each other nor both receive a bucket from each other at the same time). Therefore, at each time step they all synchronously reconsider their action on the basis of their neighbours: if both neighbours are taking the same action, they choose the opposite action, otherwise they randomise with $\frac{1}{2}$ probability of each action. Figure 2.1 depicts two configurations, where the two colours represent the two different possible actions.

(i) Which of the starting configurations (A or B) is more likely to lead to an alternating pattern, i.e., one of the goal states in which neighbouring robots are taking opposite actions?

(ii) Is there a positive probability that a process that starts as Configuration A will reach Configuration B at some time point?



(a) Configuration A.



(b) Configuration B.

Figure 2.1: Two possible configurations of Robot Bucket Brigade.

2.1 Introduction

In this chapter, we present a formal definition of a Flag Coordination Game (Section 2.2.1), and provide some examples from other domains (Section 2.2.2). We also discuss prior related work (Section 2.3) along with some background technical results we will use later in the dissertation (Section 2.4). The motivational example (Problem 1) that opens this chapter will be formalised in Section 2.2.2 and solved in subsequent chapters.

2.2 Flag Coordination Games

In this section, based on the informal description of Flag Coordination Games given in Chapter 1, we provide a detailed and formal definition of such processes. Later, we frame different well-known problems as Flag Coordination Games to better understand the potentialities and restrictions of our model.

2.2.1 Formal Rules of a Flag Coordination Game

Let $G = (V, E)$ be a graph and X be a set of colours (or states). We are interested in games in which, as time progresses, nodes may possibly change their colours. That decision is not necessarily deterministic, and thus we define the random process $\{S_t\}_{t\geq 0}$ as a family $\{S_t \mid t \in T\}$ of random variables indexed by some set T, where S_t is the colouring of vertices of G at time t.[i] Formally, each S_t is a function $S_t : V \to X$ that associates a colour $x \in X$ to a node $v \in V$. Note that when $T = \{0, 1, 2, \dots\} =: \mathbb{N}$, we have a *discrete-time* process (in which we say $t \in T$ is a *round* of this game), whereas if, for example, $T = \mathbb{R}$, we have a *continuous-time* process.[ii] We are primarily going to explore Flag Coordination Games based on discrete processes. At this point, we have not yet fully specified what defines (or might define) $\{S_t\}_{t\geq 0}$. This is done below.

The random process $\{S_t\}_{t\geq 0}$ is based on local decisions, made at each node. As previously discussed, such decisions cannot be made taking into account any other information than the current (or previous) colourings. First, we define the goal set Γ as a subset of all possible colourings of V with colours in X, i.e., $\mathcal{S} = X^V$. Also,

[i] Although counter-intuitive, we will use the general term *random process* for $\{S_t\}_{t\geq 0}$, that also includes processes that might be deterministic depending on, for example, the agents' algorithms.

[ii] Note that in Computational environments, real numbers must be approximated by rational numbers, and therefore we can consider such processes to be discrete given the enumerability of $\mathbb{Q}$. We allow $T = \mathbb{R}$ for continuous processes to be consistent with Markov chain literature.

each node $v \in V$ might not be able to see all other nodes, and thus we define the visibility function ϕ, formally $\phi_G : V \times T \to \mathbb{P}(V)$, that associates to a given node $v \in V$ and time $t \in T$ a subset of nodes that it can see at that time. We say ϕ_G formally also depends on the graph G, for example, we often define $\phi(v)$ as $\mathcal{N}(v)$, the neighbourhood of v. Moreover, we assume our agents might not have infinite memory. Indeed, most of the Flag Coordination Games studied in this dissertation assume agents have no memory of previous rounds, and so we define a function $\psi : V \times T \to \mathbb{N}$ that assigns to each node, at a given time, the number of previous rounds it can remember when making new decisions. Finally, we consider that each node v might not be able to choose between any flag (or colour) in any given round, thus we define the function $\beta : V \times T \to \mathbb{P}(X)$, which associates to pair (v, t) a subset of X of colours at v's disposal in time t.

Taking all of these functions into account, we define a set of algorithms $\mathcal{A}$ such that for each $v \in V$, there is $\alpha_v \in \mathcal{A}$ that determines v's decision on that given round. We will consider mostly randomised algorithms, although there are examples (see Example 2.2.5), in which they are deterministic.

The final component of the rules of a Flag Coordination Games is a *scheduler* σ. Possibly based on T, β, Γ, ψ, and ϕ, it determines whether, for example, nodes act synchronously, or, if not, in what order and when. Formally, $\sigma(t) = V' \subset V$, a subset of nodes that act in time t. We can also consider games in which σ is in the possession of an attacker that wants to avoid the game reaching one of the goal states, in which case we also consider how much this attacker can remember of previous rounds. Most times we are going to consider a *dumb* scheduler that makes nodes act synchronously.

We say a game $\mathcal{F}$ with starting configuration S_0 is a winning game if it eventually reaches a state $\gamma \in \Gamma$.

Definition 2.2.1 (Flag Coordination Game) *In order to summarise the definition discussed in this section, we define (the rules of) a Flag Coordination Game as a tuple* $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ *such that*

- G*: The graph* $G = (V, E)$ *in which the game takes place. It can be either a directed or undirected graph. We can also have a dynamic graph* $G(t) = (V, E(t))$*, which edges might change through time. We denote* $v \in V$ *as* ***agents****, or simply* ***nodes*** *in this game.*

- T*: The set of rounds of the game. It can be either discrete or continuous.*

- X: *The set of colours in the game. We will use the terms* ***colour****,* ***flag****, and* ***opinion*** *interchangeably. We will also refer to* $x \in X$ *as an* ***agent's current state****.*

- Γ: *The goal set. This is a subset of the colourings of nodes of* G*, i.e.* $\Gamma \subset \mathcal{S}$*, where* $\mathcal{S} = X^V$*. It depends on* G *and* X*. We assume* Γ *is known to all nodes.*

- ϕ: *The function that associates a subset of* V *to each node* v *at round* t*, i.e.,* $\phi : V \times T \to \mathbb{P}(V)$*. We say that the induced subgraph of* $\phi(v,t)$ *is the* ***visibility of node*** v *at round* t*.*

- β: *The function that associates a subset of* X *to each node* v *at round* t*, i.e.,* $\beta : V \times T \to \mathbb{P}(X)$*. We say that* $\beta(v,t)$ *is the set of* ***flags*** *(or colours) available to node* v *at round* t*.*

- ψ: *For discrete processes, the* ***memory*** *function* $\psi : V \times T \to \mathbb{N}$ *associates each node and time to the number of previous rounds it remembers, subject to its visibility in each of the previous rounds. If* $\psi(v,t) = 0$ *for all* $t \in T$*, then* v *only knows the current configuration of the game at a given time. For continuous-time games,* $\psi : V \times T \to \mathbb{R}$ *associates an agent and a particular time* $t \in T$ *to the time length that* v *remembers at a given time* t*, i.e., configurations from* $S_{t-\psi(v)}$ *up to* S_t*.*

- σ: *The function that associates each point in time (or each round) to a subset of nodes to play at that round, i.e., a* ***scheduler*** $\sigma : T \to \mathbb{P}(V)$*. This scheduler may also take into account previous or current configurations of the game.*

- $\mathcal{A}$: *The set of functions* α_v *that, for node* v*, associates round* t *to an* ***algorithm*** *that decides* v*'s colour in the next round. Functions in* $\mathcal{A}$ *might depend on* T*,* ϕ*,* Γ*,* β*,* ψ *and, most importantly, the previous configurations of this game up to the current round. We consider that algorithms may include 'no action' as a possible decision, even if the node's current colour is unknown to the node.*

Most importantly, we define $\{S_t\}_{t\geq 0}$ *as the random process* $\{S_t \mid t \in T\}$ *indexed by* T *that describes this game. Formally,* $S_t : V \to X$ *is a function that colours the nodes of* G *with colours in* X*. We usually denote* S_0 *as the initial* ***colouring****, or initial* ***state****, or initial* ***configuration*** *of a game* $\mathcal{F}$*. We will sometimes denote '****rules*** $\mathcal{F}$ ***of a Flag Coordination Game****' simply as 'Game* $\mathcal{F}$*'.*

*Moreover, we say that $S_0^t := (S_0, \dots, S_t)$ is the **trace of the game** up to round t, and that $S_0^\infty := (S_0, \dots, S_t, \dots)$ is the trace of a full game. Finally, we use $(\mathcal{F}, S_0)$ to refer to a game $\mathcal{F}$ with initial configuration S_0.*

REMARK 2.2.2. Unless otherwise stated, we are going to consider agents that are memory-less, i.e., $\psi(v,t) = 0$, for all $v \in V$, $t \in T$, for the remainder of this dissertation.

Recall that a key aspect of Flag Coordination Games is that the only information that an agent v may transmit to others (that have v in their line of vision at that given time) is their current state.

REMARK 2.2.3. We are going to denote both S and s as colourings of G, i.e., functions from V to X, with the difference that S will indicate the configuration of a game at a given time, whereas s will denote a configuration independently of a running game. That way, when we ask $\Pr(S_t = s \mid S_0)$, we mean 'probability of configuration S_t in round t being equal to state s given that the initial configuration is S_0'.

Given a set of rules of a Flag Coordination Game, we might be interested, for example, in the expected number of rounds until a goal is reached given an initial configuration denoted by $\mathbb{E}(\tau \mid S_\tau \in \Gamma)$, where $\tau = \min_t\{S_t \in \Gamma\}$, or even in the probability that a given game ends successfully, i.e., that it eventually reaches a configuration $\gamma \in \Gamma$.[iii]

2.2.2 Examples of Flag Coordination Games

We start by showing that the robot bucket brigade process presented in Problem 1 can indeed be seen as a Flag Coordination Game. Questions (i) and (ii) raised in Problem 1 are going to be answered in Chapters 3 and 5, respectively.

EXAMPLE 2.2.4 (ROBOT BUCKET BRIGADE AS FLAG COORDINATION GAME). Consider Problem 1 discussed earlier in this chapter. We are now framing it as a Flag Coordination Game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$. Here, G is a path $(v_1, \dots, v_n)$

[iii]Consider the notion of communication complexity, i.e., the number of bits needed to be exchanged between the agents to share information, and consider that $\overline{\phi}$ stands for the maximum visibility among the n agents. Then, the communication complexity is bounded by $O(\tau n \overline{\phi} \log |X|)$ bits. In other words, at each round, each agent learns the colour of at most $\overline{\phi}$ other agents, that can each be encoded with $\lceil \log |X| \rceil$ bits.

of size n, $X = \{\bullet, \bullet\} = \{\text{receive-then-pass}, \text{pass-then-receive}\}$, T is a discrete set, for example, non-negative integers. At this point, we need to clarify our choice of colours. We want our goal configurations to be stable, thus to receive buckets cannot be a colour (otherwise will have agents receiving buckets indefinitely and not passing them on). For that reason, we define receive-then-pass as the state in which agents start time t by receiving buckets to then pass them on to its neighbours before the end of time t. We define pass-then-receive analogously. Furthermore, $\Gamma = \{\gamma_1, \gamma_2\}$, where γ_1 and γ_2 are the two proper colourings of G,[iv] with $\gamma_1(v_1) = \text{receive-then-pass}$ and $\gamma_2(v_1) = \text{pass-then-receive}$. We define $\phi(v) = \mathcal{N}(v)$,[v] and $\beta(v) = X$, $\forall v \in V$. The scheduler σ is such that all nodes act synchronously. Finally, the algorithms α_v are such that, for $i \notin \{0, n\}$

$$S_{t+1}(v_i) = \begin{cases} S_t(v_{i+1}), & \text{with probability } \frac{1}{2}, \text{ and} \\ S_t(v_{i-1}), & \text{otherwise.} \end{cases} \tag{2.1}$$

Also, $S_{t+1}(v_0) = S_t(v_1)$ and $S_{t+1}(v_n) = S_t(v_{n-1})$, both with probability 1.

We can also have deterministic Flag Coordination Games, depending only on the initial state. Cellular automata are examples of discrete processes with such deterministic behaviour, having the celebrated Game of Life by John Conway as one of the best known cellular automaton.

Example 2.2.5 (Conway's Game of Life). John Conway's Game of Life [30] can be seen as an example of a Flag Coordination Game in which each new state is fully determined by the previous state. How can we, then, derive the rules of a flag coordination game that represents the Game of Life? We say G is the infinite two-dimensional grid with edges between each node and their eight neighbours[vi]. The set X has only two colours, *alive* or *dead*. Time T is discrete, $T = \{0, \dots, t, \dots\}$, and $\psi(v) = 0$ and $\phi(v, t) = \mathcal{N}(v) \cup \{v\}$ for all v and all t. All nodes have all flags available at all times, so $\beta(v, t) = X$ for all pairs (v, t). All cells act synchronously and therefore $\sigma(t) = V$ for all t. We can define the set of algorithms $\mathcal{A}$ even before defining Γ, because they are fixed. Note that algorithms take into account the current configuration S. We can represent α_v by showing what happens from one

[iv] A proper colouring of G is such that given pair of neighbouring nodes, their colours do not match.

[v] Note that here we do not even assume agents can see their current state.

[vi] Here we could have no edges at all as long as visibility of nodes is modified accordingly.

round to the next in the (not random) process S. Denote k_t as the number of alive neighbours of v in round t. We have that, for all $t \in T$ and $v \in V$,

$$S_{t+1}(v) = \begin{cases} alive, & \text{if } k_t \in \{2,3\} \text{ and } S_t(v) = alive, \\ alive, & \text{if } k_t = 3 \text{ and } S_t(v) = dead, \\ dead, & \text{otherwise.} \end{cases}$$

Finally, we define the goal set Γ. Here there are many possible end states that we might be interested in (note that most games will never end). For example, we might want to define Γ_1 as the set of states that are stable, in the sense that if $S_t \in \Gamma_1$, then $S_{n+1} = S_n$ for all $n \geq t$ with probability 1. More generally, we might want Γ_m as the set of recurrent states such that the time of first return is always m, i.e., $S_t \in \Gamma_1$, then $S_{n+m} = S_n$ for all $n \geq t$.

Next, we present an example of a game in which, although agents have complete memory of previous rounds, they are not able to see their own state at any point. In fact, their goal is precisely to find our their own state.

Example 2.2.6 (Muddy Children Problem). The commonly studied Muddy Children Problem [4] can be summarised as follows. Consider n children standing in a circle. At least one child has mud on their forehead (and all the children know this), and each child's individual task is to establish whether they are one of those with muddy foreheads by looking at the other children but with no mirror or any communication, except the following: at the end of every hour, when a common clock rings, any child that has rationally concluded that they themselves must have mud on their foreheads will immediately announce that conclusion publicly. Assuming they are all rational agents and are not malicious nor faulty in any way, how is this game going to unfold given an initial number of muddy children?

We will now frame the muddy children problem as a Flag Coordination Game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$. We can have $G = (V, E)$ as, for instance, the complete graph with n nodes, where n is the number of children in the game, T is a countable set, so we can define $T = \mathbb{N}$. The set of colours (or states) in this game is $X = \{\text{mud, no mud, mud detected}\}$. The initial configuration may have the nodes coloured with any of the two first colours, but we only allow the children the options of *mud detected* and *no action* (note that *no action* is not a colour, but the choice for the node to not change their current colour), i.e., $\beta(v) = \{\text{mud detected}\}$. We need to restrict the visibility of each agent to all other agents except themselves, so that: $\phi(v,t) = V \setminus \{v\}$, $\forall (v,t) \in V \times T$. Moreover, agents have complete memory, so $\psi(v,t) = t$ for all $v \in V$. Finally, the scheduler σ is such that all nodes act synchronously, so $\sigma(t) = V$ do all t.

Our desired algorithm for v is to wait (i.e., take no action) until round k, where k is the number of *mud* nodes that v can see. If no agent changes to *mud detected* until round k, then chose *mud detected* for round $k+1$. To be consistent with our model, we have to define a public set of goal states Γ. Because we cannot simply give away the desired configuration to the nodes on the basis of the number of *mud* coloured ones, we can define $\Gamma = \{\gamma \mid \gamma(v) \neq \text{mud } \forall v \in V\} \setminus \{\text{all mud detected}\}$. This way, the set Γ does not give the nodes any new information and prevents them from arbitrarily choosing *mud detected* in the first round, because if they all do so they are trapped in the non-winning *all mud detected* state. Moreover, the rules of this Flag Coordination Game guarantee that the game is a winning game regardless of the initial state S_0. The duration of these games under these rules is always equal to k, where k is the number of muddy children in the initial configuration.

In the next example, we state the graph proper colouring problem as a Flag Coordination Game. Note that the colouring problem is hard even in a non-distributed way, and thus this is not an attempt to solve an NP-hard problem (of finding a colour with $\chi(G)$)[vii], but rather to show that our model can describe such a problem as well. A solution for distributed colouring of graphs considering communication between agents (therefore not a Flag Coordination Game) can be found in [46].

Example 2.2.7 (Proper Colouring of Graphs). Consider a Flag Coordination Game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ played in an unidrected graph G. Agents (at each node) aim to proper colour this graph with colours in X, and acting synchronously in a discrete time set T, i.e., $\sigma(t) = V, \forall t \in T$. We may assume agents (nodes) are memory-less and have access to all colours available, $\psi(v) = 0$ and $\beta(v) = X$ for all $v \in V$ and $t \in T$, and that their visibility is only their neighbours, $\phi(v,t) = \mathcal{N}(v), \forall t \in T$. A goal configuration is one in which no neighbouring agent is coloured the same, thus $\Gamma = \{\gamma \in \mathcal{S} \mid \gamma(v) \neq \gamma(w) \text{ if } (v,w) \in E\}$. Finally, we may define set $\mathcal{A}$ such that, in round t with configuration S_t, an agent v will choose for round $t+1$ a colour at random from the set $(X \setminus \{S_t(w) \mid w \in \mathcal{N}(v)\})$. If this set is empty, they choose a random colour from X.

[vii] Notation $\chi(G)$ stands for the chromatic number of a graph, i.e., the minimum number of colours needed to proper colour graph G.

We now frame a problem studied by Vincent Blondel *at al.* in [7] as a Flag Coordination Game, where a dynamic graph $G(t)$ is modelled using the visibility function $\phi(v, t)$ evolving with time.

EXAMPLE 2.2.8 (COORDINATION BY COMPUTING AVERAGE VALUES). Let $G(t)$ be a complete finite dynamic graph with n nodes, $E(t)$ edges such that $G(t)$ is strongly connected for all $t \in T$. Let also S_0 be an initial configuration of values in $X \subset \mathbb{R}$.[viii] For the i-th node $v_i \in V$, the algorithm α_{v_i} is such that

$$S_{t+1}(v_i) = \sum_{j=1}^{n} a_{ij}(t) S_t(v_j) \tag{2.2}$$

Where $A(t)$ is a non-negative matrix with entries $a_{ij}(t)$ and T is a discrete set of time steps t. For the Equal Neighbour Model [7, Page 2], we assume that each node performs an average of the current value of all its neighbours (including its own value). Therefore, we set $\beta(v, t) = X$ for all v, t. The goal set is any consensus configuration, i.e., $\Gamma = \{\gamma_x \mid x \in X\}$, where $\gamma_x(v) = x$, $\forall v \in V$.

We finally assume that if $(i, j) \in E(t)$ infinitely often, then there is an integer B such that, for all t, $(i, j) \in E(t) \cup E(t+1) \cup \cdots \cup E(t+B-1)$. In these conditions, the agreement algorithm guarantees asymptotic consensus [7, Theorem 1].

We now discuss an example from Edsger Dijkstra's seminal paper in which he introduces a formalisation of self-stabilising systems. We have chosen this example not only for its historical importance, but also because it involves a potentially malicious agent controlling when nodes act. For now we are just stating it as a Flag Coordination Game, whereas later in Section 2.3 we will present properties of this system and background definitions.

EXAMPLE 2.2.9 (DIJKSTRA'S SELF-STABILISATION PROBLEM #1, 1974). Let a game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ be such that $G = (V, E)$ is a directed cycle of size n, i.e., there is a direct edge from v_i to v_{i+1}, for $1 \leq i < n$, and from v_n to v_1.[ix] Nodes can only see themselves and the neighbour to which there is a direct edge to, i.e., for all $t \in T$, $\phi(v, t) = \mathcal{N}(v) \cup \{v\}$. The set $X = \{0, \ldots, K\}$ is such that $K \geq n$, and $T = \mathbb{N}$ is discrete. The goal set is given by

$$\Gamma = \{\gamma \mid (\gamma(v_1) = \gamma(v_2)) \vee (\gamma(v_i) \neq \gamma(v_{i+1}), \text{ for } i \neq 1\} \tag{2.3}$$

[viii] Note that set X is countable because the number of initial nodes is finite and that they only perform averages between values of subsequent rounds.

[ix] For simplicity, in future instances in this dissertation, we are going to abuse notation by omitting $(\bmod n)$ when considering labels of nodes in a cycle.

We assume that the the scheduler σ is in the hands of a malicious agent that wants to prevent configurations in Γ from being achieved. There are, however, some restrictions from σ. Although the malicious agent has a complete memory of previous rounds, at a given round, they cannot freely choose any agent v to act in that round (note that game is then asynchronous). Instead, they can only choose, for $i \neq 1$, an agent v_i if $S(v_i) \neq S(v_{i+1})$, or v_1 if $S(v_1) \neq S(v_2)$.

Although the motivation behind these results are gong to be discussed in more detail later on, there is a set of algorithms that guarantees the agents to not leave the goal set regardless of the scheduler's choice, i.e., a set of algorithms such that if $S_{t_0} \in \Gamma$, then $S_t \in \Gamma$ for $t \geq t_0$. Note that they only act in certain conditions, and their algorithms are deterministic. These algorithms α_{v_i} are

$$S_{t+1}(v_i) = \begin{cases} S_t(v_{i+1}) & \text{if } i \neq 1 \text{, and} \\ S_t(v_2) + 1 \pmod{K} & \text{if } i = 1. \end{cases} \tag{2.4}$$

As a final example, we slightly simplify a well-known concept of spin alignment in statistical mechanics in order to understand it as a Flag Coordination Game [38].

EXAMPLE 2.2.10 (ISING MODEL). The two-dimensional Ising model for ferromagnetism (and antiferromagnetism) includes a lattice in which each site would have a positive (up) or negative (down) spin (see [50] for a book on the two dimensional model and [38] for Ising's original paper). In the ferromagnetic model, spins tend to be aligned with their neighbours whereas in the antiferromagnetic model they tend to be in opposite directions. We can model a simplified process as a Flag Coordination Game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ such that G is a (large) two-dimensional grid, $X = \{+1, -1\}$, T is continuous. For ferromagnetism, Γ is the set of two consensuses in G whereas, for antiferromagnetism, Γ represents the two proper colourings of G with two colours. Nodes can see their four neighbours and their algorithm is to choose one at random and copy (ferromanetism) or choose the opposite direction (antiferromagnetism).

As an illustration of what **is not** a Flag Coordination Game, we present the following example.

EXAMPLE 2.2.11 (COUNTEREXAMPLE: PUSH MODEL). Consider a process on a graph G in which at each round, an agent v contaminates one or all of their neighbours with v's current opinion. This is not a Flag Coordination Game because for those, we assume that each agent independently decides on their eventual changes of colour, instead of being forced to do so by another agent.

2.3 Related Work

The problem of distributed consensus in computational systems has been extensively studied, including specifically in multi-agent contexts; for reviews, see e.g., [57, 64]. If we consider communications protocols in which nodes base their decisions only on the colour of one of their neighbours (chosen at random), the probability of convergence for each colour and the complexity of the expected duration has been established by Hassin and Peleg [35, Corollary 2.2] for any non-bipartite graph.

Theorem 2.3.1 (Restatement of Hassin and Peleg, 2001) *Let G be a non-bipartite undirected graph such that nodes have a common clock and change or keep their colours by copying a neighbour uniformly at random, synchronously in rounds, until a consensus is reached. The probability of a given colour c to win that consensus game is*

$$\sum_{v \in V_x} \frac{\deg(v)}{2|E|}, \tag{2.5}$$

where V_x is the subset of nodes that are coloured x. Moreover, the time for the process to end is bounded by $\mathcal{O}(n^3 \log n)$.

Experiments with human participants for proper colouring of graphs on networks were conducted by Kearns *et al.*. They studied consensus processes in which there was no bias towards any particular colour in [42], but also processes in which participants would be paid more if, say, blue wins (although payment would only be made if a consensus was achieved) in [41]. These authors explored different restrictions on the visibilities of the participating human agents and showed that more information does not necessarily lead to better performance. This finding is in line with the well-known phenomenon in Statistics that a larger sample does not necessarily lead to more accurate conclusions.[x] There are two key differences between [42] and the Flag Coordination Games we explore in Chapters 3 and 5. First, [42] does not assume that agents share a common clock, so that agents could change their selected colours at will, asynchronously. Second, the agents in the experiments by Kearns *et al.* were actual humans who were able to use any decision algorithm or combination of algorithms, or none at all, to select colours. Real humans might also have been

[x] An infamous example goes back to 1936, when the magazine *Literary Digest* wrongly predicted the outcome of the US presidential elections despite conducting a poll in which ballots were sent out to more then 20 million residencies. *Gallup*, in contrast, correctly predicted the winner with a much smaller sample of only twenty thousand reports. For a detailed analysis of why Literary Digest's poll failed, refer to [72].

whimsical or malicious. Note that, seen as a Flag Coordination Game, the processes studies by Kearns would embed a continuous time set T, because participants could change their state at any time.

A game-theoretical approach for graph colouring was studied by Panagopoulou and Spirakis in [59]. In their model, each node v chooses a colour and then receives a payoff equal to the number of nodes that have chosen the same colour, unless a neighbour of v is one of those nodes choosing the same colour, in which case the payoff to v is zero. The authors prove that a Nash Equilibrium is always possible in this game. The key difference from our work is that Panagopoulou and Spirakis do not require nodes to choose their colours synchronously, whereas we do require this in our analysis of consensus games.

Other papers that consider different variants of the distributed consensus problem are [16, 46, 14]. In brief, in the work by Cooper *et al.* ([16]), nodes make their decisions based on two random neighbours, not just one. In [46] by Kuhn and Wattenhofer, one-round algorithms are studied instead of an evolutionary process. For Chaudhuri *et al.*, in [14], the number of available colours for the nodes is $\Delta + 2$, whereas in our work the number of colours is not a function of Δ (e.g., we use two colours in any bipartite graph for the graph colouring problem for any Δ). A social influence and consensus game model in which the population grows, and other related problems, have been described by Matthew O. Jackson in [39].

We now provide a review of Alain Sarlette's work [66, 67], which can be summarised as a study of the collective behaviour of agents in structures with high symmetry, with no hierarchy nor external interference. Several aspects of Sarlette's research overlap with our study of Flag Coordination Games. For example, both assume the visibility of different agents might vary, as well as no leader that controls the group. Moreover, we both provide a detailed analysis of processes on the circle [68]. The main difference, however, is that agents in coordination control are moving along the structure (e.g., a circle), whereas in Flag Coordination Games they are static and change their state, instead of position, in each round. Another key aspect in which Sarlette's work differs from ours is that we consider algorithms based on randomness, whereas his processes are deterministic.

Convergence in multi-agent coordination is also studied by Vincent Blondel *et al.* in [7]. Their work can also be seen as a Flag Coordination Game (see Example 2.2.8). Each node holds a value at each given time (their *flag*), which is then updated on the basis of the values of nodes that they can see at this given time. The update rule does not depend on a random decision of each node, but rather on the current (dynamic) set of edges in $G(t)$. The main difference between their work and our results in chapters to follow is that we assume nodes make a possibly random decision at each time, and we consider a finite set of choices for each agent at each time, instead of a value in $\mathbb{R}$. For models related to Blondel's, refer to work by Tsitsiklis *et al.* [74, 75, 6] and Vicsek *et al.* [76].

There is also earlier work in the theory of distributed systems which is relevant to our work.

In 1974, Dijkstra introduced a formalisation of self-stabilisation in distributed systems. His paper [24] became widely known only after a talk by Lamport in 1984, which was subsequently published as [47]. We provide the pertinent background from Dijkstra's paper in Definition 2.3.2.

Definition 2.3.2 *A* ***privilege*** *is a boolean function of the current agents' states that is given to a node v. We say a privilege is present at a given time if the function is true at that time. Dijkstra defines a global state as* ***legitimate*** *if it follows the following criterion:*

(i) in each legitimate state, one or more privileges will be present;

(ii) in each legitimate state, each possible move will bring the system again to a legitimate state;

(iii) each privilege must be present in at least one legitimate state; and

(iv) for any pair of legitimate states, there exists a sequence of moves transferring the system from the one into the other

Finally, a system is ***self-stabilising*** *if and only if, regardless of the initial state and regardless of the privilege selected each time for the next move, at least one privilege will always be present and the system is guaranteed to find itself in a legitimate state after a finite number of moves.*

In order to clarify Dijkstra's definitions, please refer to Example 2.2.9, based on the original problem #1 in [24]. In that, a privilege is present if and only if the scheduler is allowed to choose a given node to act. In other words, for $i \neq 1$, the privilege in v_i is present if $S(v_i) \neq S(v_{i+1})$, and the privilege in v_1 is present if $S(v_1) \neq S(v_2)$. We can see that, regardless of the initial configuration S_0 and the choices of the malicious agent controlling the scheduler σ under the rules described in Example

2.2.9, the game is always self-stabilising according to Definition 2.3.2. This family of games was studied with a game-theoretical approach by Apt *et al.* in [2].

We now introduce the well-known concept of Markov Decision Process (MDP) [61], in order to be able to highlight similarities and differences when compared to Flag Coordination Games.

Definition 2.3.3 (Markov Decision Process) *A Markov decision process is a tuple* $\langle S, A, T, R \rangle$

(i) $\mathcal{S}$ *is a set of states.*

(ii) X *is a set of actions*

(iii) $T(s, x, s')$ *is the state transition function and denotes the probability of moving from* s *to state* s' *on taking action* x*, with* $s, s' \in S$ *and* $x \in X$*.*

(iv) $R(s, x)$*, is the reward function, which outputs the reward of taking action* x *in state* s*, with* $s \in S$ *and* $x \in X$*.*

Although MDPs capture the idea of a group of agents aiming to jointly achieve a shared goal as in Flag Coordination Games, MDPs assume the system (or the agents) have no memory (see Example 2.2.6 for an example of a Flag Coordination Game in which agents have longer memory). Moreover, MDPs assume agents have global knowledge, which is not necessarily the case for Flag Coordination Games, in particular not for the ones studied in this dissertation. Finally, in Flag Coordination Games we assume the restriction that agents do not send messages, but their current state can be seen by the subset of agents that can see them at that given time. That might be because communication is either too expensive or the environment in which agents are located does not allow them to exchange messages.

In his work, Mihaylov [51, 52] studied decentralised coordination in multi-agent systems in great detail. In particular, he studied both pure coordination and anti-coordination processes, in which agents seek a global configuration with only local actions. The algorithm proposed by him is based on pairwise interactions between neighbours in the network for each agent to decide on their next state. A novel aspect of this model compared to the related literature in multi-agent systems, is that, in Mihaylov's algorithm, only the agent that initiates the pairwise interaction with a neighbour takes that interaction into account when choosing their next state. This algorithm always eventually reaches a global goal configuration. He assumes agents do not have knowledge of their position in the network, or the names of their neighbours. Finally, each agent has a positive probability of not interacting with any neighbour and therefore maintaining their current state for the following round.

The fact that in Mihaylov's algorithm only one of the agents change their state on the basis of their pairwise interaction can be seen in part as a particular case of a consensus game, as the one discussed in Theorem 2.3.1, but played on a directed graph instead (see Theorem 2.4.17 by [18]). With that in mind, the probability of an agent keeping their current state can be represented by a loop edge from the agent to itself.[xi] That is the reason why Mihaylov's algorithm guarantees convergence. In Chapter 3 we are going to give probabilities for convergence in situations in which agents cannot keep their own state. This might be necessary in scenarios where agents do not know their current state, or where the costs of an agent to discover its current state are prohibitive, or they are are somehow forced to renew their decision periodically. As we are going to see also in Chapter 3, both pure coordination and anti-coordination will be particular cases to be defined later as *generalised consensus* (see Definition 3.2.1).

We are also interested in the probability of convergence for each one of the possible colours in a consensus game. We will use a result by Cooper and Rivera [18] that gives us the probabilities of convergence in directed graphs given an initial configuration of colours. Morris DeGroot also studies consensus protocols using Markov chains in [22]. The restriction of those analyses is that consensus must be achieved with probability 1 as the number of steps goes to infinity, and therefore graphs that may generate state loops are not considered. This result, to be used in Chapter 3, will be reproduced in Theorem 2.4.17 once some technical background on Markov chains is introduced.

2.4 Technical Background

In this section, we provide some technical background on stochastic processes, including Markov chain, martingales, and the linear voting model. Beforehand, for completeness, we state a few definitions of graphs.

Definition 2.4.1 (Cycle Graphs) *A* ***cycle graph*** C_n *(also referred to by* ***n-cycle***, ***n-ring***, *or even* ***circle****) is an undirected graph such that* $V = \{v_1, \dots, v_n\}$

[xi] The reason why we cannot fully see Mihaylov's algorithms as a linear voting model (see Section 2.4.3) is because the probability of an agent maintaining their state depends on their previous interaction, and thus might change over time, whereas linear voting models assume probabilities are constant.

and $E = \{(v_1, v_2), (v_2, v_3), \dots, (v_n, v_1)\}$*. If* n *is odd, we will say* C_n *is an* ***odd cycle****. Otherwise,* C_n *is an* ***even cycle****. We say we move* ***clockwise*** *if we consider the sequence* $(v_1, v_2, \dots, v_n, v_1, \dots)$*. It is considering this sequence that we add or subtract indexes of nodes in a cycle. For example, in the context of* C_{20}*,* $v_{15+7} = v_2$*.*

We also make reference to ***odd nodes*** *(or* ***odd positioned nodes****) in a cycle, i.e., the set* $\{v_k \mid k \text{ is odd}\}$*. Analogously, we refer to* ***even nodes*** *(or* ***even positioned nodes****). When cycles are depicted in figures, unless stated otherwise,* v_1 *will be the top-most vertex (with indexes increasing clockwise) (see Remark 2.4.3).*

Definition 2.4.2 (Miscellaneous Graph Definition) *Define the* ***neighbourhood of a vertex****, denoted by* $\mathcal{N}(v)$*, as the set of vertices connected to it, i.e.,* $\mathcal{N}(v) = \{w \mid (v, w) \in E\}$*. We define the* ***degree of a vertex*** *is* $\deg v = |\mathcal{N}(v)|$*. A* ***m-regular graph*** *is such that* $(\forall v \in V) \deg v = m$*. A graph* $G = (V, E)$ *is* ***bipartite*** *with partitions* V_1 *and* V_2*, with* $V_1 \cup V_2 = V$ *and* $V_1 \cap V_2 = \emptyset$*, if every edge* (v, w) *is such that* v *and* w *are in different partitions.*

A ***complete graph*** K_n *is such that* $(\forall v \in V)\mathcal{N}(v) = V \setminus \{v\}$*. A* ***star graph*** *is such that* $V = \{w, v_1, \dots, v_{n-1}\}$ *and* $E = \{(w, v_i) \mid 1 \leq i < n\}$*. A* ***path*** *can be seen as a cycle with a missing edge, i.e.,* $E = \{(v_1, v_2), \dots, (v_{n-1}, v_n)\}$*. We denote path graphs as* $(v_1, \dots, v_n)$*.*

A ***dynamic graph*** $G(t) = (V, E(t))$ *is such that the set of edges might change as a function of time* t*.*

REMARK 2.4.3. Unless said otherwise, images of cycles C_n with nodes $\{v_1, \dots, v_n\}$ will depict node v_1 in the top most vertex, with indexes increasing clockwise. Images of bipartite graphs, unless stated otherwise, will depict partition V_1 as the top partition (with nodes depicted in order $v_{11}, v_{12}, \dots,$ from left to right), and V_2 as the bottom one (with nodes depicted in order $v_{21}, v_{22}, \dots,$ from left to right)

Definition 2.4.4 (In-matrix and Out-matrix of a Graph *G*) *Let* $G = (V, E)$ *be a finite digraph.*[xii] *Given some fixed order of the nodes* $V = \left\{v_1, \dots, v_{|V|}\right\}$*, the* ***(row-normalised) in-matrix*** *of* G *is the* $|V| \times |V|$ *matrix* $F := (f_{ij})$*, where*

$$\text{if } (v_j, v_i) \in E \text{ then } f_{ij} = \frac{1}{\left|v_i^-\right|}, \text{ else } f_{ij} = 0. \tag{2.6}$$

[xii] Note that we abuse notation by not distinguishing notation for edges in both undirected and directed graph. The notation $e = (v, w)$ in a digraph means that there is a directed edge from v to w, but not necessarily otherwise.

Analogously, the ***(row-normalised) out-matrix*** *of* G *is the* $|V| \times |V|$ *matrix* $H := (h_{ij})$*, where*

$$\text{if } (v_i, v_j) \in E \text{ then } h_{ij} = \frac{1}{|v_i^+|}, \text{ else } h_{ij} = 0. \tag{2.7}$$

Definition 2.4.5 (Weakly Connected Graph) *Let* G *be a digraph and let* $\widetilde{G}$ *be the undirected graph generated from* G *by replacing each directed edge in* G *by a undirected one in* $\widetilde{G}$ *(ignoring repetitions). We say that* G *is weakly connected if, and only if,* $\widetilde{G}$ *is connected.*

2.4.1 Markov Chains

Definition 2.4.6 (Markov Chain) *A sequence of random variables* $\{Y\}_{t \in T}$ *that takes values in a countable set* $\mathcal{S}$ *is said to be a Markov chain if it satisfies the Markov property, i.e., if* $\forall t \geq 1,\ s, s_0, \ldots, s_t \in \mathcal{S}$*,*

$$Pr(Y_{t+1} = s \mid Y_t = s_t, \ldots, Y_1 = s_1, Y_0 = s_0) = Pr(Y_{t+1} = s \mid Y_t = s_t) \tag{2.8}$$

A **time-homogeneous** Markov chain has the property that the transition probability from state i to state j does not depend on time. Unless stated otherwise, all Markov chains studied in this dissertation are time homogeneous.

Every time-homogeneous Markov chain on a finite set $\mathcal{S}$ can have its behaviour modelled by a transition matrix

$$P = \{p_{ij}\} \tag{2.9}$$

Where p_{ij} denotes the probability of the Markov chain transitioning from state s_i to state s_j in one step at a given time. Because every row of P sums to 1, we say that P is row stochastic. Note that then $\lambda = 1$ is an eigenvalue of P, i.e., there is $v \neq 0$ such that $Pv = v$. Therefore, $\lambda = 1$ is also an eigenvalue of P^{-1} (or, alternatively, a left eigenvalue of P). This motivates the definition of stationary distribution of a Markov chain.

Definition 2.4.7 (Stationary Distribution) *Let* P *be the transition matrix of a Markov chain. We say that* μ *is a stationary distribution of* P *if*

$$\mu P = P \tag{2.10}$$

Definition 2.4.8 *A state $s \in \mathcal{S}$ is called* ***persistent****, or* ***recurrent****, if, for some $t \geq 1$,*

$$Pr(Y_t = s \mid Y_0 = s) = 1 \tag{2.11}$$

Otherwise, the state is called ***transient****.*

We now introduce the concept of irreducibility of Markov chains.

Definition 2.4.9 (Irreducible Markov Chain) *A Markov chain is irreducible if, and only if, all states are recurrent.*

Note that the stationary distribution is unique (up to multiples) if the Markov chain is irreducible.

Definition 2.4.10 (Reachable States) *Regarding reachability, we say that a state s_i is reachable by a state s_j in a Markov chain Y if, for some $t \geq 1$,*

$$Pr(Y_t = s_i \mid Y_0 = s_j) > 0 \tag{2.12}$$

The following example, known as Gambler's Ruin, gives us the probabilities of reaching each one of two absorbing states. Informally, suppose a gambler starts with a fortune of k, $0 \leq k \leq n$. At each round, there is a probability p that it wins.

Example 2.4.11 (Gambler's Ruin). Let $Y_{t+1} = Y_t + Z_{t+1}$ be a random walk on $[0, n]$ starting at $Y_0 = k$, where $\{Z_t\}_{t \geq 1}$ forms an independent and identically distributed sequence of random variables distributed as $Pr(Z_t = 1) = p$ and $Pr(Z_t = -1) = q = 1 - p$.

Assume also that 0 and n are absorbing states, this is, if $Y_\tau = 0$, then $Y_t = 0$, $\forall t \geq \tau$. Analogously for n. Let $Pr(\tau_0 < \tau_n \mid Y_0 = k)$ be the probability for the random walk to visit 0 before visiting n when starting at position k. Then,

$$Pr(\tau_n < \tau_0 \mid Y_0 = k) = \frac{k}{n} \tag{2.13}$$

if $p = q$. And given by

$$Pr(\tau_n < \tau_0 \mid Y_0 = k) = \frac{1 - \left(\frac{q}{p}\right)^k}{1 - \left(\frac{q}{p}\right)^n} \tag{2.14}$$

if $p \neq q$.[xiii]

[xiii] For a proof, see [12, Example 6.1.3].

2.4.2 Martingales

Martingales will be useful in our analysis in both Chapters 3 and 5. For a more detailed approach, see books by Brémaud [12] and by Durrett [29]. In loose terms, a martingale is a sequence of random variables that does not tend to increase or decrease.

Definition 2.4.12 (Martingales) *Let $\{Y_t\}_{t\geq 0}$ and $\{Z_t\}_{t\geq 0}$ be two sequences of discrete real-value random variables (i.e., **real-valued stochastic processes**) such that for each $t \geq 0$*

(i) Y_t is a function of t and $Z_0^t := (Z_0, \ldots, Z_t)$, and

(ii) $\mathbb{E}(|Y_t|) < \infty$ or $Y_t \geq 0$.

We say that $\{Y_t\}_{t\geq 0}$ is a martingale with respect to $\{Z_t\}_{t\geq 0}$ if

$$\mathbb{E}(Y_{t+1} \mid Z_0^t) = Y_t. \tag{2.15}$$

A classic example of a martingale is a particular case of Example 2.4.11, as follows:

Example 2.4.13 (Fortune in a Fair Game). Let $Y_{t+1} = Y_t + Z_{t+1}$ be the fortune of a gambler in time t. At each round, she bets k into a fair game, i.e., wins or loses with equal probability. Here, the stochastic process $\{Z_t\}_{t\geq 1}$ represents the outcome at the end of round t and is such that $Pr(Z_t = k) = \frac{1}{2}$ and $Pr(Z_t = -k) = \frac{1}{2}$. Then,

$$\mathbb{E}(Y_{t+1} \mid Z_0^t) = \frac{1}{2}(Y_t + k) + \frac{1}{2}(Y_t - k) = Y_t. \tag{2.16}$$

Therefore $\{Y_t\}_{t\geq 0}$ is a martingale with respect to $\{Z_t\}_{t\geq 1}$.

Our review of stochastic processes thus far prompts the question regarding the formal definition of the stopping time of a Markov chain. In Example 2.4.11 we say that the game ends when the chain reaches either value 0 or n. However, how do we formally define the duration of a stochastic process?

Definition 2.4.14 (Stopping time) *Let $\{Y_t\}_{t\geq 0}$ be a stochastic process with values in a countable set E. Let τ be a random variable with values in $\mathbb{N} \cup \{\infty\}$. We say that τ is a Y_0^t-stopping time if for all $m \in (\mathbb{N} \cup \{\infty\})$, the event $\{\tau = m\}$ can be expressed in terms of the variables $Y_0, \ldots, Y_t$.*

Our main proofs are going to use the following result, which informally states that if a martingale is bounded, then the expected value at the stopping time is equal to the expected value at the beginning.

Theorem 2.4.15 (Corollary of Doob's Optional Sampling Theorem) *Let $\{Y_t\}_{t\geq 0}$ be a martingale with respect to $\{Z_t\}_{t\geq 0}$ and let τ be a Z_0^t-stopping time. Suppose at least one of the following conditions hold,*

(i) $Pr(\tau < k) = 1$, for some $k \geq 0$.

(ii) $Pr(\tau < \infty) = 1$ and $|Y_t| \leq K < \infty$ when $t \leq \tau$.

Then, $\mathbb{E}(Y_\tau) = \mathbb{E}(Y_0)$.

Note that Theorem 2.4.15 is a weak version of Doob's optional sampling theorem, which will not be used in this dissertation in its more general form.

2.4.3 Linear Voting Model

In this section, we briefly introduce linear voting models of Cooper and Rivera [18], to be used in our proofs mainly from Chapter 3.

Definition 2.4.16 (Linear Voting Model) *Let $G = (V, E)$ be a graph, $|V| = n$ and $\mathcal{M}$ be the set of all matrices $n \times n$ such that M is a row-stochastic matrix with, in each row, exactly one entry 1 and all the others 0. Let l be a probability distribution over matrices in $\mathcal{M}$. Finally, let S_0 be the initial colouring of G with colours in a set $X = \{0, \ldots, |X| - 1\}$, with the update rule given by*

$$S_{t+1} = M_t S_t, \tag{2.17}$$

*where M_t are independently and identically distributed matrices sampled from l. We say that this process is a **linear voting model** with parameters (l, S_0).*

The following theorem provides a solution for the consensus games on graphs as long as they always converge. Although this is the opposite of what we explore in the following chapter, we will draw from their results to establish a solution for games that fail to converge.

Theorem 2.4.17 (Cooper and Rivera, 2016) *Let* $(S_t)_{t\geq 0}$ *be a linear voting model with parameters* (l, S_0)*, mean matrix* H*, and* $X = \{0,1\}$*. Moreover,* $\Gamma = \{\gamma_0, \gamma_1\}$ *represent the set of consensus configurations in* $x = 0$ *and* $x = 1$*, respectively. Assume that* H *has a unique stationary distribution* μ *and that the time for consensus is finite, i.e.,* $\tau < \infty$*. Then,*

$$Pr(S_\tau = 1 \mid S_0) = \sum_{v\in V} \mu(v) S_0(v) \tag{2.18}$$

Note that what prevents us from using this theorem to understand failure of convergence in Flag Coordination Games is that one of its hypotheses requires convergence time to be finite.

We are also going to use one of Cooper and Rivera's results, [18, Lemma 3], that says that both synchronous and asynchronous consensus games are linear voting models.

2.4.4 Conclusion

In this section, we have identified all related work relevant to the study of Flag Coordination Games. As was seen, nobody has studied exactly the problem we consider, although we will be able to draw on the results and the methods of this other work.

Chapter 3

Flag Coordination Games: Consensus or Failure of Convergence?

PROBLEM 2 (CONSENSUS IN A CYCLE). Consider a set of twenty agents playing a Flag Coordination Game in a circle, with initial configuration as in Figure 3.1. Each node represent an autonomous agent that can decide to change their colour at the beginning of every round, choosing from a set of 3 colours. They aim to reach consensus, but can only see their neighbours. Thus, they all follow an algorithm given by: at each round, each agent chooses one neighbour at random and copies its colour. All changes are made synchronously. In these conditions, what is the chance that they eventually succeed in achieving consensus?

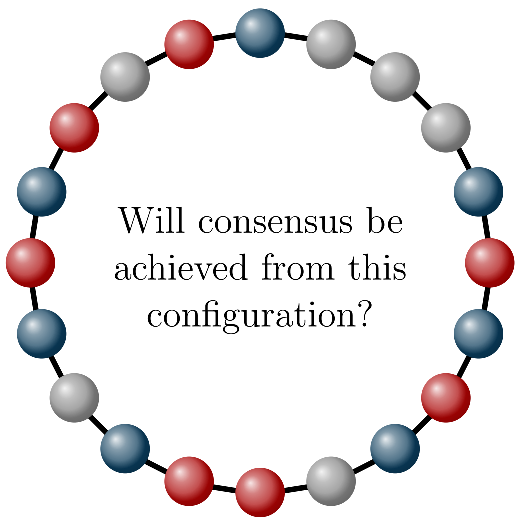


Figure 3.1: Consensus Game on a Cycle C_{20} with 3 Colours.

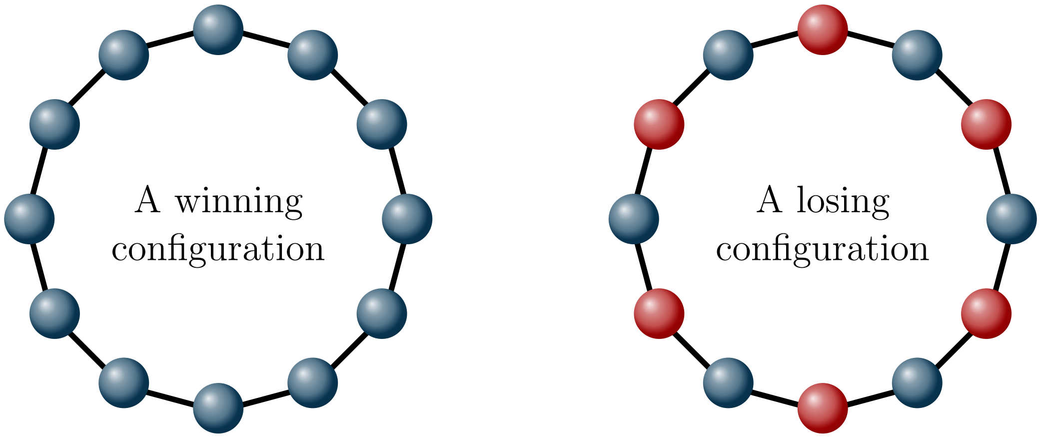


Figure 3.2: A Consensus in Blue (left) and a Configuration from which Consensus Will Never be Achieved (right) on a Cycle C_{12}.

3.1 Introduction

In this chapter, motivated by Problem 2, we will explore the situations in which a so called losing configuration might arise, and what are the probabilities involved. Note that this might happen in game described in Problem 2 if an alternating pattern is reached. We are going to define formally consensus games shortly. For now, consider a consensus game similar to the one in Problem 2, but now in a 12-cycle instead. Figure 3.2 exemplifies one situation in which consensus is achieved and one situation in which such a game is *trapped* in a loop (of size 2). At first, we are interested in the following questions

B1 Why are there losing configurations in the first place?

B2 What is the probability of each colour winning?

B3 How long will it take for either a winning or a losing configuration to be reached?

B4 Which are the initial states that might lead to a loop such as the one in Figure 3.2? Is it the case that any initial configuration that is not already in consensus can lead to a loop?

Question **B1** can be immediately answered by observing that not only consensus games on cycles of even length (such as C_{20} and C_{12}) but also games in any bipartite[i] graph will admit losing configurations for games with two or more colours. Recall that Theorem 2.3.1 excludes bipartite graphs from their analysis, and thus by solving the probability problem for bipartite graphs we will be extending Hassin and Peleg's results for any undirected graph G.

3.2 Games on Undirected Graphs

In this section, we are studying Questions **B2**, **B3**, and **B4**. In order to do that, we observe that consensus protocols in distributed systems can also be seen as Flag Coordination Games. We first define a slightly broader class of consensus games, in which not only monochromatic goal states can be achieved.

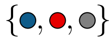
Definition 3.2.1 (Generalised Consensus Game) *Consider the tuple given by $\mathcal{F}_{GC} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$ to be the set of rules of a Flag Coordination Game played in a (non-dynamic) graph $G = (V, E)$, where $X = \{x_0, \ldots, x_{|\Gamma|-1}\}$. Also, $\Gamma = \{\gamma_0, \gamma_1, \ldots, \gamma_{|\Gamma|-1}\}$, such that, for a given pair (v, x), where $v \in V$ and $x \in X$, there exists exactly one $\gamma \in \Gamma$ with $\gamma(v) = x$. We define $\beta(v) = X$ and $\psi(v, t) = 0$ for all $v \in V$ and all $t \in T$. For undirected graphs, the visibility $\phi(v, t)$ of each vertex v is, for any $t \in T$, the set of neighbours of v, denoted by $\mathcal{N}(v)$. For directed graphs, $\phi(v, t) = v^+ := \{u \mid (v, u) \in E\}$. Finally, for each v, the algorithm α_v consists in choosing on round t a neighbour of v according to some probability function,[ii] say u, then observing which $\gamma \in \Gamma$ is the one such that $\gamma(u) = S_t(u)$. We then define the value $S_{t+1}(v) = \gamma(v)$.*

The algorithm above is well defined because, for each pair (v, x), where $x = s(v)$, there is only one goal configuration in which v takes colour x. We use the term *generalised consensus* because, assuming the nodes know where they are and which other nodes they can see, they adhere to the winning configuration that the randomly chosen neighbour belongs to. In particular, if for a given k, $0 \leq k < |\Gamma|$, $\gamma_k(v) = x_k$, $\forall v \in V$, then we have a consensus problem in the usual way. Either in usual consensus problems or in anti-consensus problems in bipartite graphs, the

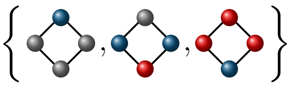
[i]Recall (Definition 2.4.2) that G is a bipartite graph with partitions V_1 and V_2, denoted by $G = (V_1, V_2, E)$, if $V_1 \cup V_2 = V$, $V_1 \cap V_2 = \emptyset$ and $\forall (u, v) \in E$, either $u \in V_1$ and $v \in V_2$ or $v \in V_1$ and $u \in V_2$.

[ii]That may possibly depend on the current colours of v and its neighbours.

assumption that nodes know their place (and of their neighbours) in the network can be lifted.

Example 3.2.2 (Generalised Consensus Game). Consider a generalised consensus game played in $G = C_4$ and such that $X =$  with the set of goal configurations given by

$$\Gamma = \{\gamma_1, \gamma_2, \gamma_3\} = \tag{3.1}$$



For example, at any point during a game, if the left most node of C_4 chooses the top most node at random and this top most node is currently blue, then, the left node will turn gray in the next round because it follows the same goal configuration as the top node is currently in, i.e., γ_1.

Note that this now generalised consensus Flag Coordination Game for any graph G does not require that agents know their *current* colour in order to make a decision. Although each agent has to make a decision of a colour at each round, this decision may be forgotten immediately afterwards, and before deciding colours at the next round.

In order to answer our questions posed in previous section, we will explore generalised consensus protocols on undirected graphs. We will focus our attention on bipartite graphs. Apart from the even-length cycles presented earlier in this work as a motivation for the study of bipartite graphs, we can also find such examples arising from competing standards in a network comprised of agents of two distinguished groups that always interact across groups, never within. For example, consider the bipartite graph G that represents doctors (partition V_1) and patients (partition V_2), in which each edge (v_1, v_2) indicates that a v_2 is a patient of doctor v_1. The same patient may consult with more than one doctor (of different specialisations), and clearly a given doctor may have more than one patient. The different colours represent different health insurance providers. We assume agents have no intrinsic preference for one provider to detriment of another to use as a patient (resp. to accept as a doctor), but they do want to share the same insurance of their doctor (resp. patient). Taking in account they are allowed one choice that can be changed from time to time, we may see this process as a Flag Coordination Game. The formal definition of Flag Coordination Games on bipartite graphs is given below.

Definition 3.2.3 (Game on Undirected Bipartite Graphs) *Let us denote by $\mathcal{F}_2 = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$ the rules of a generalised consensus flag coordination game played on an undirected bipartite graph $G = (V_1, V_2, E)$, with $V = V_1 \cup V_2$.*[iii] *We also simplify the collection of algorithms of agents in this game by setting the probability function in each α_v to be a* ***uniformly random choice*** *among the neighbours of v. We also define what is a* monochromatic *partition in a broader way, in line with Definition 3.2.1: we say partition V is monochromatic in round t if $\exists \gamma \in \Gamma$ such that $\forall v \in V_i$, $S_t(v) = \gamma(v)$. For short, we say that V is γ-monochromatic.*

Later in this section, we will define single-partition games, games in which there is only one reachable winning configuration (Proposition 3.2.10). Alternatively, these games always have a non-randomising partition: a partition whose nodes have a deterministic behaviour. In order to provide a motivation for single-partition games and the **split** function (see Definition 3.2.24), we describe an interesting connection between annihilating random walks on cycles and Flag Coordination Games.

3.2.1 Flag Coordination Games and Random Walks

Consider G an n-cycle (or n-ring), n even, and also random-walking particles each positioned at a different node of G. We consider further that there is an even number of random-walking particles in each partition of this cycle. Note that partitions in a cycle of even length are given by the set of odd nodes and the set of even nodes (recall Definition 2.4.1). At each round of this game, each particle walks clockwise or counter-clockwise with probability $\frac{1}{2}$ each. They all move synchronously. If two particles meet, both disappear. The game ends when there are no particles left. Note that particles that start within an odd distance between each other will never meet, because they are always in opposite partitions of the cycle.

Annihilating random-walking particles on a ring have been studied by Grigoriev and Priezzhev in [33]. They establish the transition probabilities between configurations of the same number of random walks, i.e., they study cases in which no pair of particles meet. For simplicity, they assume all particles lie in the same partition of a ring of even length. For a given start configuration and a final configuration, they give the transition probability of the group of particles in an arbitrary number

[iii] When depicting bipartite graphs, we will place partition V_1 as the upper partition, and V_2 as the lower one.

of discrete rounds. For other approaches on consensus and random walks on graphs, see [15].

Independently of the game described above, consider a Flag Coordination Game $(\mathcal{F}_2, S_0)$ as in Definition 3.2.3, where $X = \{\text{blue}, \text{red}\}$ and G is not only bipartite but also a cycle. For simplicity, we assume the goal states are the standard consensus configurations: all-blue and all-red. In a given round S_t, we say that a vertex is a *non-randomising node* if it has deterministic behaviour, that is, if both neighbours are currently showing the same colour (e.g., node v_1 in Figure 3.3). Otherwise, we have a *randomising node*. These nodes are going to choose blue or red with $\frac{1}{2}$ chance each (e.g., node v_4 in Figure 3.3).

We claim that we can draw a comparison between the two games described above according to the definition bellow.

Definition 3.2.4 (Placing Random Walks on a Consensus Game) *Given a consensus game $(\mathcal{F}_2, S_0)$ on a bipartite n-cycle, we define the initial places of random-walking particles by positioning them at the nodes that are randomising nodes in $(\mathcal{F}_2, S_0)$.*

Example 3.2.5 (Annihilating Random-Walking Particles on a Cycle). We can see an example in Figure 3.3. If a node is labelled with p_i, it then indicates that the random walking particle i sits on the node in round $t = 0$, and is to move clockwise or counter-clockwise with probability $\frac{1}{2}$ each direction, just before round $t = 1$.

Consider on one hand a consensus game on a cycle and, on the other hand, a sequence of moves (with possible annihilations) of random-walking particles on the same cycle. If we place the particles according to the consensus game using Definition 3.2.4, and let both processes then run independently, we will find that the expected duration of each process to finish is the same. Note that the end of a consensus game coincides with the moment when either a consensus is achieved or the process enters a loop (see Figure 3.2), whereas the process with random-walking particles game ends when all particles disappear.

In fact, there is more to the relation of both games than just their expected duration being the same. A formal definition and full analysis of their similarities will be explored and proved in Chapter 5 (Section 5.3.1.1). For now, we just want to focus our attention on the following simple fact: particles that lie in nodes belonging

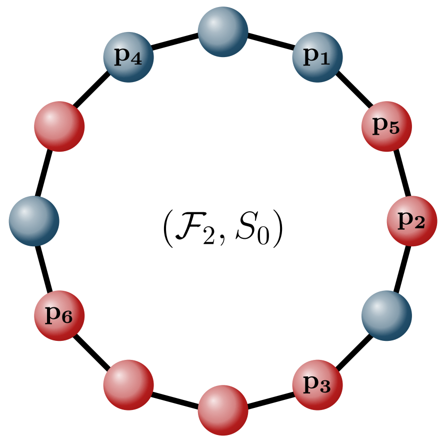


Figure 3.3: Initial states of a Flag Coordination Game, and its Correspondent Annihilating Random-Walking Particles, Depicted in the Same Graph. Nodes with p_i Indicate the Presence of Random Walking Particle i on that Node.

to different partitions at the initial round (or any given round) in the game, will never meet. That is the same to say that there are two groups of particles that are completely independent of one another. Taking Figure 3.3 as an example, particles in the group $\{p_1, p_2, p_3, p_4\}$ will never meet or have their movement interfered with particles in the group $\{p_5, p_6\}$.

Finally, note that nodes in processes of random-walking particles described in this section (to be formalised in Section 5.3.1.1) do not decide independently whether to host or not a particle in subsequent rounds, thus such processes cannot be seen as Flag Coordination Games. Even considering, instead, each particle as an agent and its state being the position in a cycle graph, agents would not be able to control whether to be annihilated or not, as it depends on the behaviour of other nodes. However, although games involving random particles do not seem to be directly suitable Flag Coordination Games, there is a clear correspondence between these two processes. That connection is what motivates us to study each partition of the graph in a Flag Coordination Game independently. We formalise this approach in the section that follows.

3.2.2 Single-partition Games

The observation that random-walking particles can only interact in the future with particles that currently lie in nodes of the same partition leads us to simplify our Flag Coordination Games in Definition 3.2.3 to games in which only one partition can have a non-deterministic behaviour. More formally, we present the following

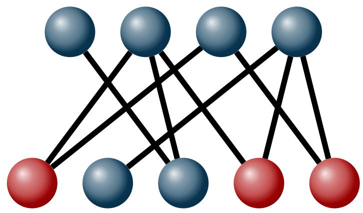


Figure 3.4: Example of a Single-partition Round.

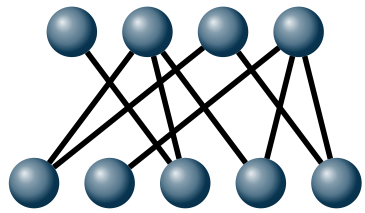


Figure 3.5: Only Reachable Consensus From Game in Figure 3.4.

definition.

Definition 3.2.6 (Single-partition round and game) *Let $(\mathcal{F}_2, S_0)$ be a general consensus game on a bipartite graph as in Definition 3.2.3. We define a **single-partition round** of $(\mathcal{F}_2, S_0)$ as a round S_t in which the behaviour of all nodes in at least one partition of G is deterministic. Moreover, we define a **single-partition game** as a game in which all rounds are single-partition rounds.*

Note that, in the case G is a cycle, the corresponding random walks model of a single-partition flag coordination game has particles in one partition only. We will now show that if round S_0 is a single-partition round, then $(\mathcal{F}_2, S_0)$ is a single-partition game.

Proposition 3.2.7 *Let $(\mathcal{F}_2, S_0)$ be a general consensus game on a bipartite graph as in Definition 3.2.3 where G is connected. If S_t is a single-partition round, then there is at least one partition, say V_1, that is monochromatic.*

Proof. Let V_2 be the non-randomising partition on such round. Then, for each $v \in V_2$, all $u \in \mathcal{N}(v)$ are coloured according to the same $\gamma_u \in \Gamma$. Because G is connected, all $u \in V_1$ must be coloured according to the same γ_u (otherwise there would be a $v \in V_2$ with neighbours coloured according to two different γ, which is not possible). We call that common colouring γ. Then, V_1 is γ-monochromatic. ■

Example 3.2.8 (Single-partition Round). Figure 3.4 depicts an example of a single-partition round of a consensus game on a bipartite graph. Note that the top partition (V_1) is blue-monochromatic, and therefore nodes on the bottom partition (V_2) are non-randomising nodes in the current round. Figure 3.5 represents the only winning state reachable from game in 3.4.

Proposition 3.2.9 *A game that eventually reaches a single-partition round has all its subsequent rounds also single-partition. In particular, if S_0 is a single-partition round, $(\mathcal{F}_2, S_0)$ is a single-partition game.*

Proof. Say V_1 is γ-monochromatic partition in a single-partition round S_t. Then, in round S_{t+1}, all nodes in Γ will have been adhered to γ, thus V_2 will be γ-monochromatic and so S_{t+1} is also a single-partition round. By induction on t, $(\mathcal{F}_2, S_0)$ is a single-partition game. ∎

Does this proposition imply anything regarding the possible final configurations of single-partition games? Indeed, the next corollary of Proposition 3.2.9 shows that there is only one possible winning state for such games.

Corollary 3.2.10 (Ending of Single-partition Games) *Let $\gamma \in \Gamma$ be such that there is a γ-monochromatic partition on the initial round of a single-partition game $(\mathcal{F}_2, S_0)$. Then, in the case the game reaches consensus (it might not), such consensus must be γ.*

We now define a function that labels edges in single-partition rounds according to whether the colour of the nodes it connects belong to the same colouring or not. This will help us keep track of how close the given single-partition game is from its only possible winning configuration.

Definition 3.2.11 (Edge-colouring Function) *Let $(\mathcal{F}_2, S_0)$ be a single-partition consensus game on a bipartite graph as in Definition 3.2.6 and $\mathcal{S}_E = X^E$ be the collection of all $|X|^{|E|}$ possible colourings (ou labellings) for the edges in G. Assume* ***wlog*** *that partition V_1 is γ-monochromatic. We define $f : \mathcal{S} \to \mathcal{S}_E$, $f(s) = r$ as the function that colour each edge $e = (u, v)$ ($u \in V_1$ and $v \in V_2$) according to whether they are currently belong to the same γ (black edge) or not (green edge), i.e.,*

$$r(e) = \begin{cases} \text{black}, & \text{if } (\exists \gamma \in \Gamma)[s(u) = \gamma(u) \wedge s(v) = \gamma(v)] \\ \text{green}, & \text{otherwise.} \end{cases} \tag{3.2}$$

In other words, in a consensus game between blue and red in which there is a blue-monochromatic partition, an edge is black if and only if the current colours of the nodes it links agree (i.e., both blue because one partition is already blue), otherwise the edge is coloured green. Note that a game ends successfully when all edges are black (and therefore blue wins). We can now give the probability of success based on the initial configuration of a single-partition game. We first formally define what we mean by “success”.

Definition 3.2.12 (Winning Game) *We say that a game* $(\mathcal{F}_2, S_0)$ *is* ***successful****, or it is a* ***winning game****, if it reaches and indefinitely stays in one of the goal states*[iv]*. In other words, a game is successful if there exists* $\gamma \in \Gamma$ *and* $\tau \geq 0$ *s.t.* $\forall t \geq \tau$, $S_t = \gamma$*, where* $(S_0, \ldots, S_t, \ldots)$ *is the trace of such a game. For* $\gamma \in \Gamma$ *and* τ *as above, we then denote* $Pr(S_\tau = \gamma \mid S_0)$ *as the probability that opinion* γ *wins* $(\mathcal{F}_2, S_0)$*, i.e., that it is eventually achieved. More generally, we define* $Pr(S_\tau \in \Gamma \mid S_0)$ *as the probability that* $(\mathcal{F}_2, S_0)$ *is a successful (or winning) game regardless of which consensus it reaches at the end, as long as a consensus is achieved.*

Definition 3.2.12 prompts us to explore what are the situations in which consensus is not achieved. Moreover, thus far τ is not yet fully defined in the sense that τ might not exist for games that never reach a consensus state. To address these issues, we first define a random variable that counts the number of labelled edges during a given game.

Definition 3.2.13 (Black Edges Counter) *Let* $(\mathcal{F}_2, S_0)$ *be a single-partition consensus game on a bipartite graph* $G = (V, E)$ *as in Definition 3.2.6. We define* $(Y_t)_{t>0}$ *as the random variable that counts the number of black edges in* S_t *according to Definition 3.2.11.*

Note that, for single-partition games on connected graphs, if $Y_0 = |E|$, then $S_0 \in \Gamma$ and therefore the game is certainly a winning game. On the other hand, if $Y_0 = 0$, then Y_1 is also null and indeed $Y_t = 0$ for $t \geq 0$. We can show this by induction. Assume $Y_t = 0$. Then, there is one partition, say V_1, that is γ-monochromatic in S_t. Therefore, V_2 will be γ-monochromatic on round S_{t+1} and also no node in V_1 will keep their colour, i.e., $S_{t+1}(u) \neq \gamma(u)$ for $u \in V_1$, because no node in V_1 on round S_t has a neighbour in γ (since $Y_t = 0$). Thus, $Y_{t+1} = 0$ and, by induction, the game will never fully reach colouring γ .

Definition 3.2.14 (Duration of a Game) *For games of the form* $(\mathcal{F}_2, S_0)$*, we now define the duration* τ *of the trace of a game being the smallest* t *such that* $Y_t \in \{0, |E|\}$*. In other words, we are considering both winning and losing games in our definition of duration. We define* $\tau_{(\mathcal{F}_2, S_0)} := \mathbb{E}(\tau \mid Y_\tau \in \{0, |E|\}, S_0)$ *to be the expected duration of a game with set of rules* $\mathcal{F}_2$ *and initial configuration* S_0*. We will also just use* τ *for* $\tau_{(\mathcal{F}_2, S_0)}$ *when clear from the context.*

[iv] Note that $(\mathcal{F}_2, S_0)$ is not necessarily a single-partition game.

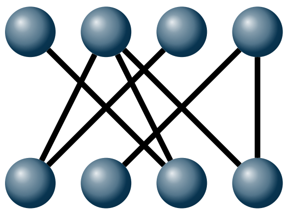


Figure 3.6: Example of a Winning Configuration.

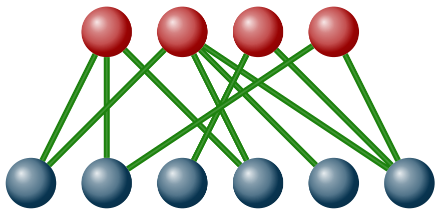


Figure 3.7: Example of a Losing Configuration.

Definition 3.2.15 *From this point on, we are more commonly going to refer to consensus in blue or red than to a general consensus in γ. We then define γ_{blue} and γ_{red} as being the states in which all nodes are coloured blue and red, respectively. More generally, we denote s_x as the state in which all nodes are coloured $x \in X$.*

REMARK 3.2.16. We will often abuse notation and use Pr(●) when referring to $Pr(S_\tau = \gamma_{\text{blue}} \mid S_0)$ and Pr(●) when referring to $Pr(S_\tau = \gamma_{\text{red}} \mid S_0)$, when clear from the context. We will also use ●, ●, and ● to refer to its respective colours.

We are now ready to answer Question **B4**: indeed, any configuration in a bipartite graph that is not already in consensus has a positive probability of non-convergence. This can be seen because there will be at least one blue edge and one black edge in G (otherwise consensus or a losing configuration would have been achieved).

EXAMPLE 3.2.17 (WINNING AND LOSING CONFIGURATIONS). Here, as usual, we assume $\Gamma = \{\gamma_{\text{blue}}, \gamma_{\text{red}}\}$. Figure 3.6 depicts a winning scenario for colour blue, in which state γ_{blue} has been achieved. Figure 3.7, on the other hand, shows an example of a game that will never reach consensus from this current state.

Theorem 3.2.18 (Winning Probabilities For Single-Partition Games) *Let $(\mathcal{F}_2, S_0)$ be a single-partition game on a connected graph G as in Definition 3.2.6. Assume, **wlog**, that partition V_1 is γ-monochromatic, for $\gamma \in \Gamma$, in S_0. Then the probability of success of $(\mathcal{F}_2, S_0)$ is given by:*

$$Pr(S_\tau = \gamma \mid S_0) = \frac{Y_0}{|E|} \tag{3.3}$$

Note that this result is similar to the one by Hassin and Peleg [35] (Theorem 2.3.1) , but now instead of considering the entire graph, we consider only partition V_1. Then, defining $(V_1)_\gamma = \{u \in V_1 \mid S_0(V_1) = \gamma(u)\}$ we have the alternative formula

$$Pr(S_\tau = \gamma \mid S_0) = \sum_{u \in (V_1)_\gamma} \frac{\deg u}{|E|}. \tag{3.4}$$

Proof. We first prove that $(Y_t)_{t\geq 0}$ is a bounded martingale (recall Definition 2.4.12) with respect to $(S_t)_{t\geq 0}$ (note that by knowing S_t we also have $r_t = f(S_t)$). Denote also $Z_t(v) = Y_{t+1}(v) - Y_t(v)$, where $Y_t(v)$ denotes the number of black edges connected to v on round t. Note that $\deg v$ stands for the number of neighbours of v.

If $Y_0 = |E|$, then $Pr(S_\tau = \gamma \mid S_0) = 1$. On the other hand, $Pr(S_\tau = \gamma \mid S_0) = 0$ if $Y_0 = 0$. Else, we call, V_t the monochromatic partition on round t. Then,

$$\begin{aligned}
\mathbb{E}(Y_{t+1} \mid S_t) = \mathbb{E}\left(\sum_{v\in V_i}(Y_t(v) + Z_t(v)) \mid S_t\right) &= \sum_{v\in V_i} Y_t(v) + \sum_{v\in V_i}\mathbb{E}(Z_t(v) \mid S_t) = \\
&= Y_t + \sum_{v\in V_i}\Big[Pr\{S_{t+1}(v) = S_t(v)\}(\deg v - Y_t(v)) \\
&\quad + Pr\{S_{t+1}(v) \neq S_t(v)\}(-Y_t(v))\Big] = \\
&= Y_t.
\end{aligned}$$

The last step follows from the fact that

$$Pr\{S_{t+1}(v) = S_t(v)\} = \frac{Y_t(v)}{\deg v} \text{ and } Pr\{S_{t+1}(v) \neq S_t(v)\} = \frac{\deg v - Y_t(v)}{\deg v}. \quad (3.5)$$

Therefore, $(Y_t)_{i\geq 0}$ is a martingale with respect to $(S_t)_{i\geq 0}$. Since $0 \leq Y_t \leq |E|$, the martingale is also bounded and thus we can apply (a corollary of) Doob's Optional Sampling Theorem (recall Theorem 2.4.15) to get $\mathbb{E}(Y_0) = \mathbb{E}(Y_\infty) = Y_\tau$, where τ stands for the duration of the game. Note that there are two absorbing states for Y: 0 and $|E|$. Thus,

$$Y_0 = \mathbb{E}(Y_0) = \mathbb{E}(Y_\infty) = |E|\, Pr(Y_\tau = |E|) + 0(Pr(Y_\tau = 0)). \quad (3.6)$$

This concludes the proof. ■

Note that there is only one winning state in a single-partition game: the state which the nodes on the randomising partition are in. Therefore $Pr(S_\tau \in \Gamma \mid S_0) = Pr(S_\tau = \gamma \mid S_0)$.

We now present an upper-bound for the expected time τ. Here we look into a formula that takes explicitly into account the number of edges of G. That result will be then explored in particular cases such as cycles and paths.

Theorem 3.2.19 (Upper-bound for Expected Duration $\mathbb{E}(\tau)$) *Let the game $(\mathcal{F}_2, S_0)$ be a single-partition game on a connected graph G as in Definition 3.2.6,* *where $|V| = n$ and $|E| = m$. If $Y_0 = 0$ or $Y_0 = m$, then the duration of the game is zero. Otherwise, let $\gamma \in \Gamma$ be the colouring of the monochromatic partition in this initial state. Denote $Y_t(v)$ as the number of black edges connected to v on round t. Finally, let V_t be the monochromatic partition on round t. Then, we have*

$$mY_0 - Y_0^2 = \mathbb{E}\left(\sum_{t=0}^{\infty}\sum_{v\in V_t} Y_t(v)\left(\deg v - Y_t(v)\right)\right) \tag{3.7}$$

Thus, because the internal sum is greater than or equal to 1 *for the duration of the game we can show that the expectation of the duration of the game $(\mathcal{F}_2, S_0)$ until there are either no black edges left (the game is a losing game) or only black edges left (colouring γ wins) is bounded by:*

$$\tau_{(\mathcal{F}_2,S_0)} \leq mY_0 - Y_0^2 \tag{3.8}$$

The proof of this theorem is a direct application of the following three lemmas. All proofs are presented after Lemma 3.2.22.

Lemma 3.2.20 $\mathbb{E}(Y_\infty^2) = mY_0$.

Lemma 3.2.21 *For each $t \geq 0$, we have*

$$\mathbb{E}(Y_{t+1}^2) - Y_0^2 = \sum_{i=0}^{t}\mathbb{E}(Z_i^2). \tag{3.9}$$

Lemma 3.2.22 *For each $i \geq 0$ we have that*

$$\mathbb{E}(Z_i^2) = \mathbb{E}\left(\sum_{v\in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)\right). \tag{3.10}$$

Proof (of Lemma 3.2.20). From $Pr(Y_\tau = m) = \frac{Y_0}{m}$, we get

$$\mathbb{E}(Y_\infty^2) = m^2 Pr(Y_\tau = m) + 0^2 Pr(Y_\tau = 0) = mY_0. \qquad \blacksquare$$

Proof (of Lemma 3.2.21). Define $Z_i = Y_{i+1} - Y_i$, i.e., the change in the total number of black edges from round i to round $i+1$.[v] It is then clear that $Y_{i+1} = Y_i + Z_i$. Note that Z_i is the sum of $Z_i(v)$ for all nodes v in one given partition of G. By Theorem 3.2.18, $\mathbb{E}(Z_i \mid S_i) = 0$. Then,

$$\mathbb{E}(Y_{i+1}^2 \mid S_i) = \mathbb{E}(Y_i^2 + 2Y_iZ_i + Z_i^2 \mid S_i) = Y_i^2 + \mathbb{E}(Z_i^2 \mid S_i)$$

By induction we have the result. $\blacksquare$

[v]Please note the change in index notation.

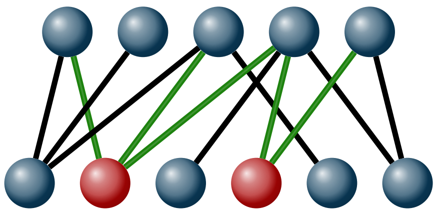


Figure 3.8: Game $(\mathcal{F}_2, S_0)$ as in Example 3.2.23.

Proof (of Lemma 3.2.22). We start by $\mathbb{E}(Z_i^2 \mid S_i)$. Recall that $Z_i(v) = Y_{i+1}(v) - Y_i(v)$. Since $\mathbb{E}(Z_i \mid S_i) = 0$, then $\mathbb{E}(Z_i^2 \mid S_i) = \mathrm{Var}(Z_i \mid S_i)$. The random variables $Z_i(v)$ are independent, then

$$\mathrm{Var}(Z_i \mid S_i) = \sum_{v \in V_i} \mathrm{Var}\left(Z_i(v)\right) = \sum_{v \in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)$$

because we have $\mathrm{Var}(\delta_i(v)) = (-Z_i(v))^2 \frac{\deg v - Z_i(v)}{\deg v} + (\deg v - Z_i(v))^2 \frac{Z_i(v)}{\deg v}$. Using $\mathbb{E}(Z_i^2) = \mathbb{E}(\mathbb{E}(Z_i^2 \mid S_i)) = \mathbb{E}(\mathrm{Var}(Z_i \mid S_i))$, we get

$$\mathbb{E}(Z_i^2) = \mathbb{E}\left(\sum_{v \in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)\right). \tag{3.11}$$

Which concludes the proof of this Lemma. Please note that the random variable $\sum_{v \in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)$ is non-negative, therefore we can apply the monotone convergence theorem to interchange summation and expectation, .i.e.,

$$\sum_{i=1}^{\infty} \mathbb{E}\left(\sum_{v \in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)\right) = \mathbb{E}\left(\sum_{i=1}^{\infty} \sum_{v \in V_i} Y_i(v)\left(\deg v - Y_i(v)\right)\right). \tag{3.12}$$

■

Example 3.2.23. Consider the initial configuration of a game $(\mathcal{F}_2, S_0)$ depicted in Figure 3.8. Here $X = \{●, ●\}$ and $\Gamma = \{\gamma_{\text{blue}}, \gamma_{\text{red}}\}$. Observe that $|E| = 12$ and $Y_0 = 7$, therefore, by Theorem 3.2.18, $Pr(●) = Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{7}{12}$, and, as expected, and because of Proposition 3.2.10, $Pr(●) = Pr(S_\tau = \gamma_{\text{red}} \mid S_0) = 0$.

Also, by Theorem 3.2.19, the expected duration of the game is bounded by $\tau_{(\mathcal{F}_2, S_0)} \leq 84 - 49 = 35$.

3.2.3 General bipartite graphs

The previous results were somehow restrictive as we assume that one entire partition is monochromatically coloured. In this section, to be able to solve the problem for an arbitrary initial configuration, we define a function that splits the original problem into two single-partition games. Then, based on the results of the two new games, we can fully determine what happens on the original one. The idea is also motivated by the relationship between consensus games on a cycle and process involving random-walking particles in the same cycle discussed in Section 3.2.1: we may know that there are two independent groups of particles that represent the change in colours of the nodes in the correspondent Flag Coordination Game. However, how do we conclude the overall result by just knowing what happens in each one?

Definition 3.2.24 (Split function) *We let $(\mathcal{F}_2, S_0)$ be a game on a connected graph G and $\boldsymbol{split}_G$ be the function that takes a colouring $s \in \mathcal{S}$ and outputs two colourings*[vi] *$\rho, \sigma \in \mathcal{S}$ such that one colouring copies the colours of s in partition V_1 and where the other colouring copies colours in V_2, colouring the remaining nodes according to the same given winning colouring $\gamma \in \Gamma$. Formally, we define*

$$\begin{gathered}\boldsymbol{split}_G : \mathcal{S} \times \Gamma \to \mathcal{S} \times \mathcal{S} \\ \boldsymbol{split}_G(s, \gamma) = (\rho, \sigma)\end{gathered}$$

Where $\rho \restriction_{V_1} = s \restriction_{V_1}$ and $\rho \restriction_{V_2} = \gamma \restriction_{V_2}$, also $\sigma \restriction_{V_2} = s \restriction_{V_2}$ and $\sigma \restriction_{V_1} = \gamma \restriction_{V_1}$.[vii]

Example 3.2.25. Figure 3.9 shows us an example of function **split** applied to the initial configuration of a game $(\mathcal{F}_2, S_0)$ resulting on the two initial configurations of the independent games $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$. As usual, $X = \{●, ●\}$ and $\Gamma = \{\gamma_{\text{blue}}, \gamma_{\text{red}}\}$. More formally, $\mathbf{split}_G(S_0, \gamma_{\text{blue}}) = (\rho_0, \sigma_0)$.

Note that the split function is solely a concrete way to visualise the independence of the behaviour of the two partitions in such games.

We are now in a position to answer Question **B2**, by stating the more general theorem for bipartite graphs, irrespective of whether the initial configurations involved are single-partition or not.

Theorem 3.2.26 (Consensus Probability in Bipartite Graphs) *Let $(\mathcal{F}_2, S_0)$ be a Flag Coordination Game as in Definition 3.2.3 and let $(\rho_0, \sigma_0) = \boldsymbol{split}_G(S_0, \gamma)$, where $\gamma \in \Gamma$ is any given winning configuration. In these conditions,*

$$Pr(S_\tau = \gamma \mid S_0) = \sum_{u \in (V_1)_\gamma} \frac{\deg u}{|E|} \sum_{v \in (V_2)_\gamma} \frac{\deg v}{|E|}. \tag{3.13}$$

---

vi Recall that both notations s and S represent an element in $\mathcal{S}$, with the contextual difference that S indicates a random variable, whereas s denotes the values the random variable S can take. However, for simplicity, regarding ρ and σ, we are not differentiating between configurations that are part of a process of not.

vii Let $f : A \to B$. We define $f \restriction_{\widetilde{A}} : \widetilde{A} \to B$, where $\widetilde{A} \subset A$ as a function that is only defined in a subset of A and coincides with f for any $\widetilde{a} \in \widetilde{A}$. We say that $f \restriction_{\widetilde{A}}$ is the restriction of f to $\widetilde{A}$

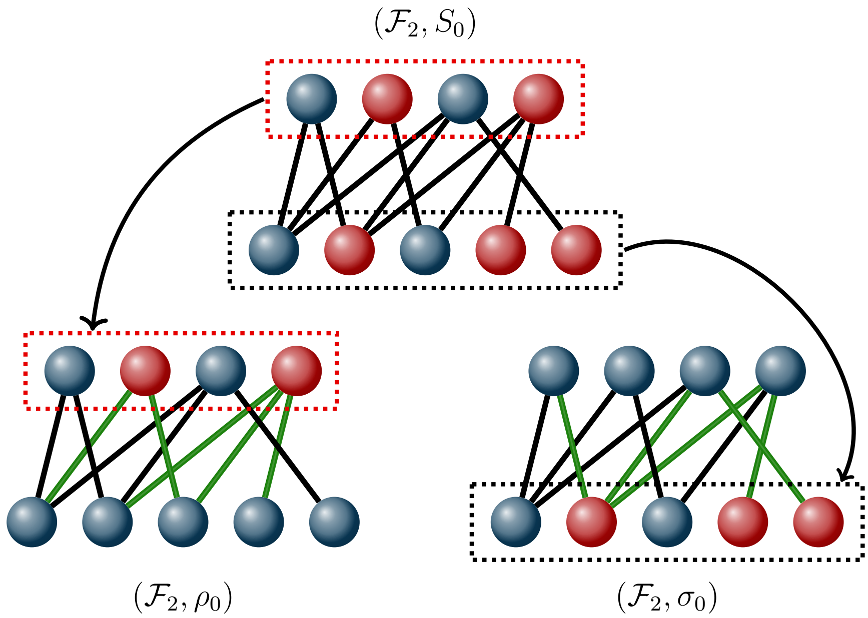


Figure 3.9: Example of Game $(\mathcal{F}_2, S_0)$ Being Split in $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$.

This comes from the fact that $Pr(S_\tau = \gamma \mid S_0) = Pr(\rho_\tau = \gamma \mid \rho_0) \times Pr(\sigma_\tau = \gamma \mid \sigma_0)$. *In other words, goal* γ *is the winning configuration of* $(\mathcal{F}_2, S_0)$ *if and only if both* $(\mathcal{F}_2, \sigma_0)$ *and* $(\mathcal{F}_2, \rho_0)$ *are winning games (note that, according to Proposition 3.2.10, such games can only reach one winning configuration, and that is* γ*). Alternatively, denoting* Y_0 *and* X_0 *as the number of black edges in* $(\mathcal{F}_2, \sigma_0)$ *and* $(\mathcal{F}_2, \rho_0)$*, respectively, then* $Pr(S_\tau = \gamma \mid S_0) = \frac{Y_0 X_0}{|E|^2}$.

Proof. The idea of the proof is straightforward: the behaviour of nodes in V_2 in $(\mathcal{F}_2, \rho_0)$ is the same as the ones in V_2 in $(\mathcal{F}_2, S_0)$. That is because a node $v \in V_2$ see the same set of colours in both games. At the same time, the behaviour of V_1 in $(\mathcal{F}_2, \sigma_0)$ is the same as V_1 in $(\mathcal{F}_2, S_0)$. During the next round, the same is true but now for the opposite partitions. Moreover, all nodes v in non-randomising partitions (the ones looking at vertices all in γ) will have a deterministic behaviour: to choose the colour $\gamma(v)$.

The core of the proof relies on the fact that the behaviour of the nodes in a given partition, say V_1, in game $(\mathcal{F}_2, S_0)$ on even rounds will never depend on decisions these same nodes took on previous odd rounds. That happens because bipartite graphs have no cycles with odd length. All the 'information' contained in partition V_1 on S_t is captured by partition V_2 on S_{t+1}, and only by partition V_2. That information will go back to V_1 on S_{t+2}. Therefore, the **split** function captures that behaviour by generating two independent games whose nodes in randomising partitions make decisions as nodes in V_1 and V_2 on $(\mathcal{F}_2, S_0)$ do. ■

In some cases we might not be interested in the final consensus but solely whether the game ends successfully (see Problem 1 for an example). In these conditions, because winning colourings are stable, we have

$$Pr(S_\tau \in \Gamma \mid S_0) = \sum_{\gamma \in \Gamma} Pr(S_\tau = \gamma \mid S_0). \tag{3.14}$$

We have so far found an analytic solution for the probability of consensus being achieved for each goal state, as well as for the game to be a losing game, for both single-partition and general games on bipartite graphs. Regarding the expected duration of games, we provided an upper bound for the process to end for single-partition games only. The next step is to generalise upper bound results for general games on bipartite graphs, as well as finding lower bounds for the expected time for both processes to end.

Both single-partition games generated by **split** function are independent. Also, the general game ends when the second single-partition game ends. However, we cannot just take the greater of the two bounds for $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$ to estimate the bound for $(\mathcal{F}_2, S_0)$. As an illustration of this, consider the problem of expected times in dice tossing: although the expected number of tosses to get a face, say "4", in one die for the first time is 6, the expected number of rounds, on the other hand, for two dice (both tossed in each round) in order to get the first "4" in both, not necessarily at the same time, is $\frac{96}{11}$ which is greater than 6.

We present the next result as a corollary of Theorem 3.2.19, as we are only combining the bounds of each single-partition game using that $\mathbb{E}(\max\{X, Y\}) \leq \mathbb{E}(X) + \mathbb{E}(Y)$. This, together with Theorem 3.2.28 below, gives us a satisfactory answer to Question **B3**.

Corollary 3.2.27 (of Theorem 3.2.19) *Let $(\mathcal{F}_2, S_0)$ be a Flag Coordination Game as in Definition 3.2.3 and let $(\rho_0, \sigma_0) = \textbf{split}_G(S_0, \gamma)$. Let Y_0 and X_0 be the number of black edges in the initial round of $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$, respectively. Then, the expected duration τ of $(\mathcal{F}_2, S_0)$ is bounded by*

$$\mathbb{E}(\tau) \leq m(Y_0 + X_0) - (Y_0^2 + X_0^2) \tag{3.15}$$

Theorem 3.2.28 (Lower-bound for Expected Duration $\mathbb{E}(\tau)$) *For a single-partition game, a lower bound for the duration τ of the game that is given by*

$$\mathbb{E}(\tau) \geq \frac{8(mY_0 - Y_0^2)}{mn} - 1 \tag{3.16}$$

Proof. For this proof we are going to use some Graph Theory results, in particular concerning the first Zagreb index [34, 20]. The first Zagreb index, $M_1(G)$, is defined by the sum of the squares of all degrees in a given graph, i.e.,

$$M_1(G) = \sum_{v \in V} \deg^2(v). \tag{3.17}$$

The particular result used here is the one by Zhou [79, Theorem 1] that, when applied to a bipartite graph G, noting that bipartite are triangle-free graphs, gives us the following bound

$$M_1(G) \leq mn \tag{3.18}$$

Other results on the first Zagreb index can be found in the work by de Caen [21], and, more recently, by Das [19]. Although Zhang and Zhou [78] presents results only for bipartite graphs, they do not aim to provide tighter bounds. Instead, they find the set of bipartite graphs of a given number of edges and nodes such that their first Zagreb index is maximised. For a recent survey on Zagreb indices, see [11].

We now focus our attention back to finding a lower bound for $\mathbb{E}(\tau)$. We start from the right-hand side of Equation 3.7.

$$\mathbb{E}\left(\sum_{t=0}^{\infty}\sum_{v \in V_t} Y_t(v)\left(\deg v - Y_t(v)\right)\right) \leq \mathbb{E}\left(\sum_{t=0}^{\infty}\sum_{v \in V_t} \frac{\deg^2(v)}{4}\right) \tag{3.19}$$

The inequality comes from the fact that any function $f(x) = x(k-x)$ defined on $x \in \mathbb{R}$ admits its maximum at $x = \frac{k}{2}$. We now apply Equation 3.18, noting that all nodes are added every two rounds, to get

$$\mathbb{E}\left(\sum_{t=0}^{\infty}\sum_{v \in V_t} \frac{\deg^2(v)}{4}\right) \leq \frac{1}{4}\mathbb{E}\left(\sum_{t=0}^{\lceil \frac{\tau}{2} \rceil} mn\right) \leq \frac{mn(\mathbb{E}(\tau)+1)}{8}. \tag{3.20}$$

Note that the $(+1)$ term is necessary for when τ is odd. ■

Corollary 3.2.29 *Let $(\mathcal{F}_2, S_0)$ be a Flag Coordination Game as in Definition 3.2.3 and let $(\rho_0, \sigma_0) = \boldsymbol{split}_G(S_0, \gamma)$. Let Y_0 and X_0 be the number of black edges in the initial round of $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$, respectively. Then, the expected duration τ of $(\mathcal{F}_2, S_0)$ is bounded below by*

$$\mathbb{E}(\tau) \geq \frac{4m(Y_0 + X_0) - 4(Y_0^2 + X_0^2)}{mn} \tag{3.21}$$

	Single-partition games on bipartite graphs	General games on bipartite graphs
Winning probability $Pr(S_\tau = \gamma \mid S_0)$	$\dfrac{Y_0}{\lvert E\rvert}$	$\dfrac{Y_0 X_0}{\lvert E\rvert^2}$
Upper-bound for expected duration $\mathbb{E}(\tau)$	$mY_0 - Y_0^2$	$m(Y_0 + X_0) - (Y_0^2 + X_0^2)$
Lower-bound for expected duration $\mathbb{E}(\tau)$	$\dfrac{8(mY_0 - Y_0^2)}{mn} + 1$	$\dfrac{4(m(Y_0 + X_0) - (Y_0^2 + X_0^2)}{mn}$

Table 3.1: Summary of Results of This Chapter for Undirected Graphs.

Remark 3.2.30 (A Note on Complexity). All games seen so far end, on average, in $O(n^3 \log n)$ rounds. That upper-bound on τ was given by Hassin and Peleg [35] for all consensus games on non-bipartite graphs, which can trivially expanded to include bipartite ones as well.

Table 3.1 summarises the result we have seen so far in this chapter, including our answers to Questions **B2** and **B3**. Results take into account that every node knows the position of the neighbours they see in the graph G. If we relax that condition determining that nodes do see the colours of their neighbours, but not their labels, then we cannot solve the generalised consensus problem in the same way. In a non-bipartite graph, the standard consensus problem can be solved, as shown in [35]. Moreover, in bipartite graphs, not only can the standard consensus problem be solved, but also the proper colouring problem. Nodes do not have to know the partition they are in nor the labels of the nodes whose colours they are looking at, as long as they know they are in a bipartite graph and whether they seek standard consensus or proper colouring of the graph. That is the case because for both problems all neighbours of a given node are coloured the same in each of the goal states $\gamma \in \Gamma$.

Solution to Problem 2. We now return to the problem posed in Problem 2. Because it is a cycle, this graph is regular (see Figure 3.1). Thus, the influence of each node is the same. Moreover, G is bipartite. Figure 3.10 rearranged nodes in Figure 3.1 evidencing the two partitions of G.[viii] It may have seemed counter-

[viii]Note that the blue node on the top left corner of Figure 3.10 corresponds to the top node in the cycle in Figure 3.1, and it is connected to the red node on the bottom right corner.

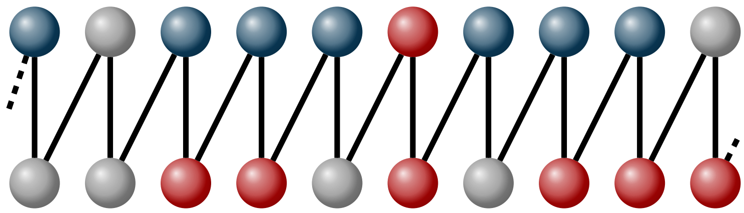


Figure 3.10: Alternative Display of the Cycle in Figure 3.1 Evidencing Partitions of G.

intuitive at first, but we can now clearly see that the probability of blue being the winning colouring, although there are 7 blue nodes of the 20 nodes in total, is zero. Note that there is no blue node in V_2 (bottom partition) of G in Figure 3.10. By Theorem 3.2.26,

$$Pr(\textcolor{blue}{\bullet}) = Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{14}{20} \times \frac{0}{20} = 0$$

$$Pr(\textcolor{red}{\bullet}) = Pr(S_\tau = \gamma_{\text{red}} \mid S_0) = \frac{2}{20} \times \frac{12}{20} = 0.06$$

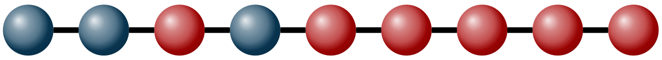
$$Pr(\textcolor{gray}{\bullet}) = Pr(S_\tau = \gamma_{\text{gray}} \mid S_0) = \frac{4}{20} \times \frac{8}{20} = 0.08$$

Thus, the probability that consensus is achieved, regardless of which, is 0.14. Note that the least common colour (also the colour with the fewest number of edges connected to nodes of that colour) is the most likely to win. Still, the most likely outcome is not success, but that the game is a losing game, with probability 0.86.

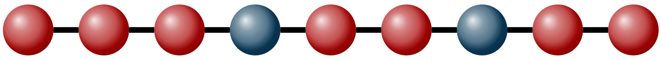
Such unexpected situations do not occur when G is non-bipartite: in these cases the most connected colour (considering the weights of edges) always has the highest probability of winning [35]. Also, the fact that non-bipartite graphs have at least one odd cycle implies that every generalised consensus game on such graphs is a winning game.

Note that our results extend the work of Hassin and Peleg (Theorem 2.3.1) to bipartite graphs. Moreover, such results propose a solution for the generalised consensus problem (see Definition 3.2.1), provided nodes are aware of their neighbours' labels and of the graph structure.

Finally, we revisit Problem 1 (Robot Bucket Brigade) presented at the start of Chapter 2 and provide a solution for its first question. The solution to its second question will be presented in Chapter 5. Note that this is an example of an anti-consensus game in which agents seek to proper colour this graph.



(a) Configuration A'.



(b) Configuration B'.

Figure 3.11: Translation of Robot Bucket Brigades Configurations Into Consensus Games.

Solution to Problem 1 (Take 1). Recall that we had a line of autonomous robots in a bucket brigade aiming to choose an action (colour) different from their neighbours', i.e., playing an anti-consensus game. In Example 2.2.4, we formally defined their goal set $\Gamma = \{\gamma_1, \gamma_2\}$ as the set of both alternating colours in this path, i.e., the two proper colourings of this bipartite graph. We are therefore under the assumptions of Definition 3.2.1 where the graph is bipartite, so we can apply Theorem 3.2.26. Before, however, we translate this anti-consensus game into a consensus game $(\mathcal{F}_2, S_0)$ to help us better visualise the goal states. In Figure 3.11, a node is blue if and only if the colour of this node in Figure 2.1 corresponds to its colour in γ_1 (recall that $\gamma_1(v_1) =$ orange). We then have, for configuration A,

$$Pr(\gamma_1 \text{ is achieved}) = Pr\left(S_\tau = \gamma_{\text{blue}} \mid S_0 = A'\right) = \frac{1}{8} \times \frac{4}{8} = \frac{4}{64} \tag{3.22}$$

$$Pr(\gamma_2 \text{ is achieved}) = Pr\left(S_\tau = \gamma_{\text{red}} \mid S_0 = A'\right) = \frac{7}{8} \times \frac{4}{8} = \frac{28}{64} \tag{3.23}$$

Therefore, the probability of agents to succeed with starting configuration A is of $\frac{1}{2}$. For configuration B, we have

$$Pr(\gamma_1 \text{ is achieved}) = Pr\left(S_\tau = \gamma_{\text{blue}} \mid S_0 = B'\right) = \frac{2}{8} \times \frac{2}{8} = \frac{4}{64} \tag{3.24}$$

$$Pr(\gamma_2 \text{ is achieved}) = Pr\left(S_\tau = \gamma_{\text{red}} \mid S_0 = B'\right) = \frac{6}{8} \times \frac{6}{8} = \frac{36}{64} \tag{3.25}$$

Therefore, the probability of agents to succeed with starting configuration B is of $\frac{5}{8}$. Thus, a game that starts at configuration B has a higher probability of reaching some generalised consensus than configuration A.

A more detailed translation from consensus to anti-consensus games can be found in Chapter 4, Section 4.4.2.1.

3.2.4 A Small Generalisation

Results presented so far in this chapter assume that neighbours are chosen with equal probability. In fact, results do not change significantly if we allow edges to have different weights in a way that the probability of v copying the colour of a given neighbour w is the weight of the edge (v, w) divided by the sum of weights of edges of the form (v, u), for $u \in \mathcal{N}(v)$. Note that this would require that both ends of a given edge are affected the same way.

Definition 3.2.31 (Weighted Edges of G) *Given a graph $G = (V, E)$, we define the function* $\boldsymbol{weight} : E \to \mathbb{R}^+$*, that associates each edge with a value* $\boldsymbol{weight}(e)$*, $e \in E$. We also extend this definition for sets in the usual way. Let $F \subset E$ be a subset of edges of G. Then,* $\boldsymbol{weight}(F) = \sum_{e \in F} \boldsymbol{weight}(e)$.

We can now generalise Theorem 3.2.18 to take weighted edges into consideration when calculating probabilities of consensus in single-partition games.

Corollary 3.2.32 (of Theorem 3.2.18) *Let $(\mathcal{F}_2, S_0)$ be a single-partition game on a connected graph G with weighted edges as in Definition 3.2.6. Assume,* ***wlog****, that partition V_1 is γ-monochromatic, for $\gamma \in \Gamma$, in S_0. Let $(\widetilde{Y}_t)_{t \geq 0}$ be the random variable that sums the weighted of black edges in round t. Then the probability of success of $(\mathcal{F}_2, S_0)$ is given by:*

$$Pr(S_\tau = \gamma \mid S_0) = \frac{\widetilde{Y}_0}{|E|} \tag{3.26}$$

Proof. This proof is nearly a copy of proof of the original Theorem 3.2.18. We just need to replace Y_t by $\widetilde{Y}_t$, and to state that $\widetilde{Y}_t$ is bounded because $0 \leq \widetilde{Y}_t \leq$ **weight**(E). ■

The main restriction of this generalisation is clear: both nodes at different ends of a given edge must apply the same weight to each other, although the sum of weights of all edges connected to each of them might be different. How do we further generalise this by allowing an asymmetric relationship between neighbouring agents? The answer to this and other generalisations will be presented in the following section.

3.3 Games on Directed Graphs

In this section, we will seek to try to understand what is behind the approaches and strategies defined in the previous section and how to generalise them. In sum, we are looking into the following questions:

C1 Bipartite graphs in generalised consensus games as seen so far in this chapter might lead to loops of states that will never lead to consensus. These loops have length 2. Is there any set of games in a graph G that might enter in a loop of size 3 instead? If so, are they tripartite graphs? How would these graphs be defined?

C2 What would be a characterisation of graphs that admit state-loops of given size?

C3 If the situation in Question **C2** happens in a graph G and initial state S_0, is there an analogous version of our **split** function that would generate 3 (instead of 2) single-partition games from S_0?

C4 In these conditions, what is the probability of generalised consensus games that admit such loops to be winning games?

C5 Are there graphs in which losing games might not include loops?

C6 Finally, what is the probability of success of these more general games?

The answer for the questions above boils down to the algorithm use by the agents, as well as the graph they are in. In order to answer Question **C1**, consider the example below in which, although there are only two possible consensus states, neither one can be achieved.

Example 3.3.1 (The Direct 3-Cycle). Consider consensus game on a digraph $G = (V, E)$ in which $V = \{v_1, v_2, v_3\}$ and $E = \{(v_1, v_2), (v_2, v_3), (v_3, v_1)\}$. Assume directed edges represent nodes' visibilities. In this example, v_1 only sees v_2, v_2 only sees v_3, and v_3 only sees v_1. Assume the initial configuration of this game is $S_0(v_1) = S_0(v_2) =$ blue and $S_0(v_3) =$ red. Considering that they uniformly at random choose a colour they see (in this case, only one choice for each), then we can see that this game is already in a 3-state loop and will never reach consensus.

The example above would have similar behaviour if our graph G had three partitions, V_1, V_2, and V_3 such that all edges go from a node in partition i to a node in partition $i+1$.[ix] Now that we understand that directed graphs (or digraphs) are to be explored, we formally define the terms of Flag Coordination Games explored in this section.

Definition 3.3.2 (Generalised Consensus in Directed Graphs) *We define a game $(\vec{\mathcal{F}}, S_0)$, where $\vec{\mathcal{F}} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ as in Definition 3.2.1, with the difference that G is a weakly connected digraph and that nodes follow a different set of algorithms $\mathcal{A}$. For each node $v \in V$, α_v is such that v copies the colour of one neighbour according to H, the row-normalised out matrix of G (see Definition 2.4.4).*[x] *The intuition here is that the i^{th} node $v_i \in V$ has a probability $h_{ij} > 0$ to copy the colour of v_j when $(v_j, v_i) \in E$.*[xi]

3.3.1 Strongly Connected Graphs

Problem 3 (Consensus in a Strongly Connected Digraph). Consider the generalised consensus game $(\vec{\mathcal{F}}, S_0)$, in a digraph G and initial configuration S_0 as depicted in Figure 3.12. Node v_{ij} is the j^{th} node in partition i. The out-matrix H is given by

$$H = \left(\begin{array}{cc|ccc|cccc}
0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} \\
\hline
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}\right)$$

Note that H has its entries representing nodes first regarding its partition, then their position within the partition. In this case: $v_{11}, v_{12}, v_{21}, v_{22}, v_{23}, v_{31}, v_{32}, v_{33}, v_{34}$. What is the probability of consensus in each of the colours involved?

For Problem 3, we cannot apply Theorem 2.4.17 by Cooper and Rivera [18] because

[ix] As usual, we abuse notation by not making explicit that a node in partition V_3 connects to a node in V_1, instead of to the inexistent V_4.

[x] In the case a node v has no out-degree, v maintains its initial colour during all subsequent rounds.

[xi] Similarly to Section 3.2.4, simple generalisation methods apply here. We are going to leave nodes' choices to be uniformly random for now.

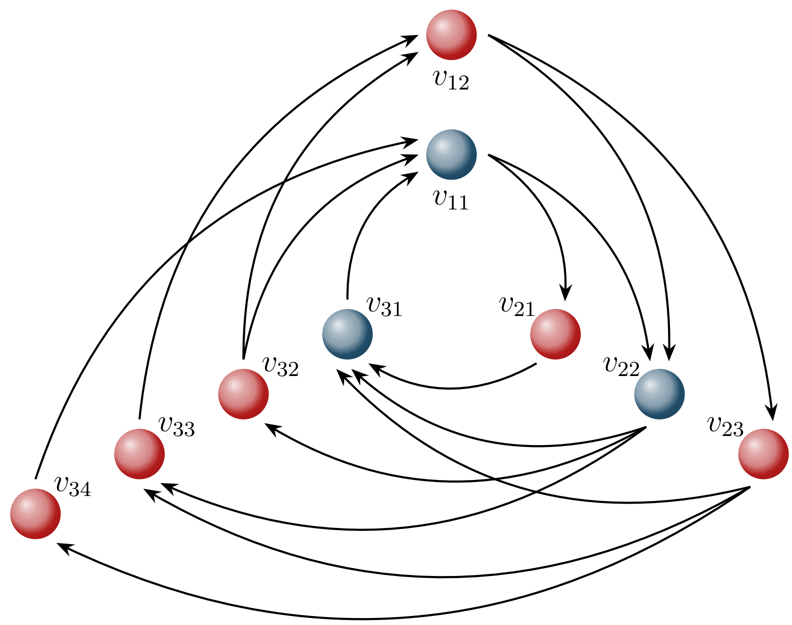


Figure 3.12: A Generalised Consensus Game $(\vec{\mathcal{F}}, S_0)$ in a Digraph G that Might Not Lead to Consensus.

the game might not have a finite duration. In order to see that, just consider the situation in which we have monochromatic partitions but of different colours. The game then will stay in a 3-state loop. We now focus our attention on the length of cycles in Figure 3.12. It is not the case that all cycles have length 3, as some have length 6, for example $(v_{12}v_{23}v_{34}v_{11}v_{22}v_{32}v_{12})$. In order to answer Question **C3**, we present two definitions and a proposition, that addresses whether a given game has the chance of entering in a loop based on lengths of cycles in G.

Definition 3.3.3 (Greatest Common Divisor of Cycles Lengths in *G*) *Let G be a digraph and $C \subset \mathbb{N}$ be the set of the lengths of all cycles in G. We then define $k = k(G) := \gcd C$, the greatest common divisor of the lengths of all cycles in G.*

Definition 3.3.4 (Digraphs that are *k*-partite) *We say that a directed graph G is k-partite if partitions $V_1, \ldots, V_k$ of V are such that every edge $(v, u) \in E$ connects $v \in V_i$ to $u \in V_{i+1}$ for some i.*[xii]

We begin addressing Question **C2**, of how to characterise graphs regarding to the possible size of state-loops that can occur if consensus games are played on these graphs, by presenting the following proposition.

[xii] Again, as usual, consider $V_{k+1} = V_1$.

Proposition 3.3.5 *If k is the greatest common divisor of the lengths of all cycles in a graph G, then G is k-partite.*

Proof. Let $v \in V$. For all $w \in V$, we define the partition that w belongs to by taking the $x(\bmod k)$, where x is the length of any path from v to w.

We show that this is well defined. First, the existence of such a path is guaranteed by the strongly connectivity of G. Also, the lengths of all paths from v to w must coincide, modulo k. If not, by concatenating both paths to the same returning path from w to v, we would have created two cycles from v to v that differ in length, modulo k (by assumption, all cycles must be $0(\bmod k)$).

Thus, by defining V_{i+1} as the set of vertices such that their distance (modulo k) from a given $v \in V$ is i, we construct partitions of V as required. ■

Note that the converse of Proposition 3.3.5 does not necessarily hold. The definition of k-partite graphs allows us to combine partitions into, say, groups of two. This way a graph with $k(G) = 6$ can be seen as a 3-partite graph if we combine each of the original 6 partitions (given by Proposition 3.3.5) into three: $V_1 \cup V_4$, $V_2 \cup V_5$, and $V_3 \cup V_6$.

Definition 3.3.6 (Generalised Consensus in k-partite Digraphs) *We define a set of generalised consensus games $(\vec{\mathcal{F}}_k, S_0)$ as in Definition 3.2.1 with the restriction that the greatest common divisor of all cycles in G is k. We also know from Proposition 3.3.5 that G is a k-partite graph.*

Question **C2** is now fully resolved with the Lemma that follows.

Lemma 3.3.7 *A consensus game $(\vec{\mathcal{F}}_k, S_0)$ on a strongly-connected digraph $G = (V, E)$ reaches consensus with probability 1 for all initial configurations if and only if $k = 1$. More generally, $(\vec{\mathcal{F}}_k, S_0)$ might only enter a state-loop of size k, otherwise it reaches consensus.*

Proof. ($\Leftarrow$) Assuming $k = 1$. Then, given an initial configuration, a game has already reached consensus or it has not. If it has, the problem is solved. If not consider $v \in V$ coloured according to some $\gamma \in \Gamma$. We note that $\gcd C_v = 1$, where C_v is the set of the lengths of the cycles passing through v. This follows from the fact that G is strongly connected. We can then show that there is a large enough $n_0 > 0$ such that for any $n \geq n_0$, we have $Pr(S_n(u) = \gamma \mid S_0) > 0$ for all $u \in V$. For that it is enough to show that there is finite n_0, such that for every $n \geq n_0$

there is a directed path from v to u of length n.[xiii] The existence of such n_0 follows from Lemma 2.1 of [65]. Thus, if the game runs long enough, it will reach (some) consensus with probability 1.

($\Rightarrow$) We now want to prove that if the game reaches consensus with probability 1, then $k = 1$. We are going to prove this by showing that if $k > 1$, then there is a positive chance that the game never reaches consensus. By Proposition 3.3.5, G is k-partite. We now observe that, if the game reaches a configuration in which one partition is all γ-monochromatic and another is $\widetilde{\gamma}$-monochromatic, for $\widetilde{\gamma} \neq \gamma$ consensus will never be reached. Thus it can not be reaching consensus for sure from all possible initial configurations. We will later show that having monochromatic partitions are not the only counterexample. In fact, for $k > 1$, any initial configuration that differs from consensus, even slightly, has a positive probability of never achieving it. ■

At this point, in order to answer Question **C3**, we informally introduce a generalisation of single-partition games for digraphs. Note that the introduction of single-partition games is not strictly necessary for our future theorems; however it assists in the visualisation of the independence of partitions in directed graphs. In sum, a single-partition game in the context of directed graphs is, as expected, a game in which nodes of all but one partition (at most) have deterministic behaviour. For such consensus games, we can apply the result by Cooper and Rivera [18] (Theorem 2.4.17) to this 'moving' partition in which nodes are randomising. There are $|\Gamma|$ possible end states for this game: in each one, the randomising partition becomes γ-monochromatic for a different $\gamma \in \Gamma$, and consensus is achieved depending on the colour of the other partitions involved.

We are not going formally to define single-partition games. However, this idea is going to be used in the proof of Lemma 3.3.12, which gives us the probabilities of consensus being achieved for each γ in a strongly connect graph G.

Definition 3.3.8 (The Influence of a Node) *Let G be a strongly connected k-partite directed graph and H be its (row-normalised) out-matrix. Let the* row *vector μ denote the* ***stationary distribution*** *of H, i.e. μ satisfies $\mu H = \mu$ (see Definition 2.4.7). We consider that μ is normalised such that its entries sum to k. We then define $\mu(v)$ as the* ***influence*** *of v, for $v \in V$. Finally, let $U \subset V$. Then, we define the influence of U as*

$$\mu(U) = \sum_{v \in U} \mu(v) \tag{3.27}$$

[xiii] Note that the colour changes run according to the reverse path.

REMARK 3.3.9. Note that we normalise μ such that the sum of its entries sum to k, instead of 1. Let us show that $\mu(V_i) = 1$ for each partition V_i. Noting that H is row-stochastic and that h_{vw} is only positive if w is in a consecutive partition of the one that v is in, we have

$$\mu(V_i) = \sum_{w \in V_i} \sum_{v \in V_{i-1}} \mu(v) h_{vw} = \sum_{v \in V_{i-1}} \left(\mu(v) \sum_{w \in V_{v-1}} h_{vw} \right) = \mu(V_{i-1}) \tag{3.28}$$

REMARK 3.3.10 (G IS STRONGLY CONNECTED IFF μ IS UNIQUE). Note that an out-matrix H is irreducible if, and only if, G is strongly connected. This means looking at H as a transition matrix of a Markov chain, all states are reachable (with positive probability) from all other states, which is clear from the strong connectivity of G. We now apply a standard result [12, Theorem 7.2.5] that states that a irreducible Markov chain is positive recurrent if, and only if, there exists a stationary distribution. Also by the same theorem, μ is unique (up to scalar multiples).

Intuitively, the higher the influence of a node v, the more it contributes to the probability of the game reaching consensus in v's current colour. The independence of partitions in such games adds another layer of complexity to the analysis, therefore in introduce the following definition before Lemma 3.3.12 (which responds Question **C4**).

Definition 3.3.11 *Let $(\mathcal{F}, S_0)$ be a Flag Coordination Game and $\gamma \in \Gamma$. Then, we define $\Theta^\gamma(S_t)$ as the sum of influences of nodes coloured according to γ at round t, i.e.,*

$$\Theta^\gamma(S_t) = \sum_{\substack{v \in V \\ S(v) = \gamma(v)}} \mu(v). \tag{3.29}$$

If G is k-partite, we analogously define $\Theta_i^\gamma(S_t)$ as the sum of influences of nodes in partition V_i that are coloured according to γ at round t. For convenience, we will use simply Θ^γ and Θ_i^γ for $\Theta^\gamma(S_0)$ and $\Theta_i^\gamma(S_0)$, respectively.

Lemma 3.3.12 *Let $(\vec{\mathcal{F}}_k, S_0)$ be a game as in Definition 3.3.6. In these conditions,*

$$Pr(S_\tau = \gamma \mid S_0) = \prod_{i=1}^{k} \Theta_i^\gamma \tag{3.30}$$

Proof. We use a similar approach to the one in Theorem 3.2.26 and apply Theorem 1 of [18] (see Theorem 2.4.17). Note that the state of vertices of V_{i-1} in round $t+1$, depends only in the state of vertices of V_i in the round t. We can then consider k parallel consensus games on k copies of G, where in the i-th consensus game we set the initial state of the vertices in V_i to their original initial state in the consensus game, but set the state of all other vertices according to the goal state γ. Denote by p_i the probability of the i-th consensus game reaching a γ winning state. We can then conclude that $Pr(S_\tau = \gamma \mid S_0) = \prod_{i=1}^{k} p_i$.

We are left to show that $p_i = \Theta_i^\gamma$. For that end, over the i-th consensus game define the random variable $X_t = \Theta_j^\gamma(S_t)$, where $j = t + i + 1 (\mathrm{mod}\, k)$. For simpicity, we introduce the boolean variable $\widetilde{S}_t(v) := 1$ if $S(v) = \gamma(v)$ and 0 otherwise. We show that the process $(X_t)_{t\geq 0}$ is a martingale with respect to the sequence S_t. We need to show that $\mathbb{E}(X_{t+1} \mid S_t) = X_t$. By linearity of expectation $\mathbb{E}(X_{t+1} \mid S_t) = \sum_{v\in V_{j+1}} \mu(v)\mathbb{E}(\widetilde{S}_{t+1}(v) \mid S_t)$. Note that

$$\mathbb{E}(\widetilde{S}_{t+1}(v) \mid S_t) = \sum_{u\in V_j} h_{vu}\widetilde{S}_t(u) \tag{3.31}$$

and, by changing the order of summation, we get that:

$$\mathbb{E}(X_{t+1} \mid S_t) = \sum_{u\in V_j} \widetilde{S}_t(u) \sum_{v\in V_{j+1}} \mu(v)h_{vu}. \tag{3.32}$$

Due to stationarity of μ and the fact that h_{vu} is non-zero only for $v \in V_{j+1}$, we have that $\sum_{v\in V_{j+1}} \mu(v)h_{vu} = \mu(u)$, which implies that $\mathbb{E}(X_{t+1} \mid S_t) = X_t$.

Now, we use (a corollary of) Doob's Optional Stopping Theorem (recall Theorem 2.4.15) together with the fact that $0 \leq X_t \leq \mu(V) = k$ to get

$$\mathbb{E}(X_0) = \mathbb{E}(X_\infty \mid X_0) = \mu(V_i)p_i \tag{3.33}$$

and this proves, using $\mu(V_i) = 1$, that $p_i = \Theta_i^\gamma$, which concludes the result. ■

Note that the theorem giving probabilities for bipartite graphs (Theorem 3.2.26) is just a particular case of the result presented above in which each edge of the undirected graph is replaced by two directed edges (one in each direction), and so the gcd of all cycles is 2. That flexibility will later allow us to find a better generalisation compared to the one in Section 3.2.4.

We can now go back to Problem 3.

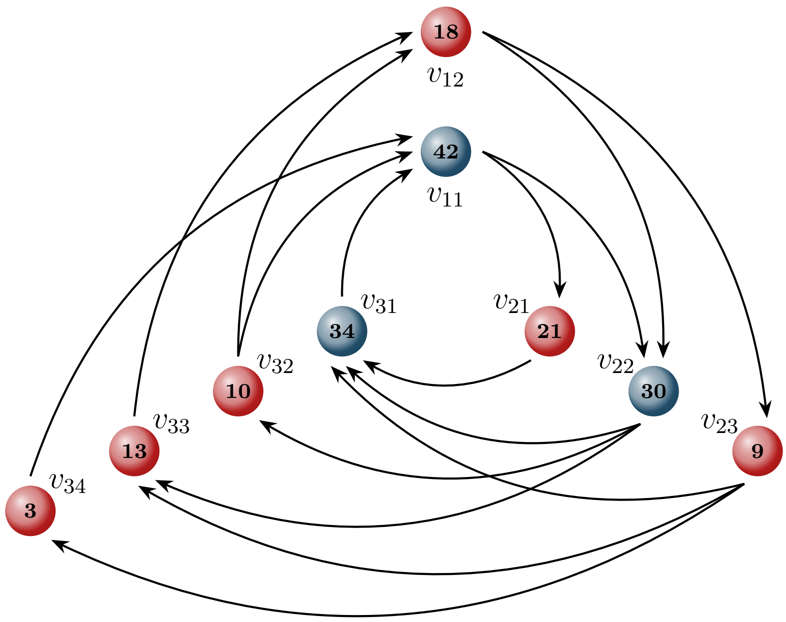


Figure 3.13: Game $(\vec{\mathcal{F}}_3, S_0)$ with Influences of Each Node (Multiplied by 60 for Readability).

Solution to Problem 3. Note that G is 3-partite with partitions $V_1 = \{v_{11}, v_{12}\}$, $V_2 = \{v_{21}, v_{22}, v_{23}\}$, and $V_3 = \{v_{31}, v_{32}, v_{33}, v_{34}\}$. We calculate the ($k$-normalised) stationary distribution μ of H to get

$$\mu = \frac{1}{60}(\underbrace{42, 18}_{V_1}, \underbrace{21, 30, 9}_{V_2}, \underbrace{34, 10, 13, 3}_{V_3}) \tag{3.34}$$

Refer to Figure 3.13 for a copy of G with influences highlighted in each node. We now apply Lemma 3.3.12 to get

$$Pr(\bullet) = Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{42}{60} \times \frac{30}{60} \times \frac{34}{60} \approx 0.20$$

$$Pr(\bullet) = Pr(S_\tau = \gamma_{\text{red}} \mid S_0) = \frac{18}{60} \times \frac{30}{60} \times \frac{26}{60} \approx 0.06$$

3.3.2 Weakly Connected Graphs

We have so far only explored strongly connected graphs in our analysis of synchronous consensus games. In this section, we extend our results to any weakly connected graph (see Definition 2.4.5). The particularity of graphs that are not strongly connected is that some nodes might not influence the final outcome of a

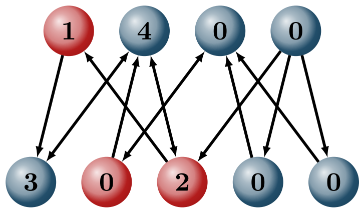


Figure 3.14: A Game $(\vec{\mathcal{F}}, S_0)$ on a Weakly Connected Graph.

consensus game in any way. For example, in a star graph in which all edges point towards the central node v, the consensus will be solely determined by v's initial colour. Recall that, according to Definition 3.3.2, a node with no out-degree maintains its initial colours for the entirety of the game.

PROBLEM 4 (CONSENSUS IN A WEAKLY CONNECTED DIGRAPH). Consider the game $(\vec{\mathcal{F}}, S_0)$ depicted in Figure 3.14. The stationary distribution of its out-matrix H is given by

$$\mu = \frac{1}{5}(\underbrace{1,4,0,0}_{V_1},\underbrace{3,0,2,0,0}_{V_2}). \tag{3.35}$$

What is the probability of each opinion to win?

Recall that in the previous section we had a unique stationary distribution μ because matrix H was irreducible (coming from the fact that G was strongly connected). However, the graph in Problem 4 is not strongly connected. How can we know that we have found the correct stationary distribution for such cases? Does having $\mu(v) = 0$ imply that v's initial colour does not influence the game at all?

We will be able to fully understand how games in weakly connected graphs behave by looking at the condensation graph of G.

Definition 3.3.13 (Condensation Graph of a Graph G) *Let $G = (V, E)$ be a digraph. Its **condensation** is the digraph $(\mathcal{K}, \mathcal{E})$ such that $\mathcal{K} \subseteq \mathbb{P}(V)$ is the set of strongly connected components (SCCs) of G and $(K, K') \in \mathcal{E} \subseteq \mathcal{K}^2$ iff $[(\exists v \in K)\,(\exists u \in K')\,(v,u) \in E$ and $K \neq K']$. A **source component** is a component with no in-degree. A **sink component** is a component with no out-degree.*

Note that nodes with no out-degree form a SCC by themselves.

EXAMPLE 3.3.14 (CONDENSATION GRAPH). Consider the game from Problem 4 in graph G. Figure 3.15 represents the condensation graph of G.

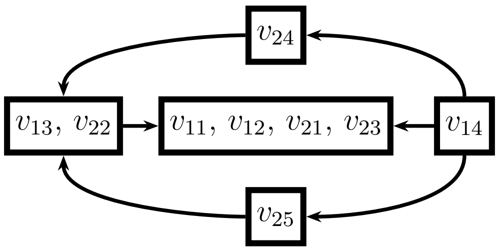


Figure 3.15: Condensation Graph of Graph in Figure 3.14.

Example 3.3.14 hints that the sink SCC of G is the one that determines the result, since all other nodes direct or indirectly depend on what happens in partition $\{v_{11}, v_{12}, v_{21}, v_{23}\}$. In the case that this partition reaches consensus, the rest of the graph will follow eventually. This is evidenced by μ and the fact that $\mu(v) = 0$ if $v \notin \{v_{11}, v_{12}, v_{21}, v_{23}\}$.

We are left with a final problem: what happens if there is more than one sink SCC? Intuitively, we can see that, because they are independent, we would need them all to reach consensus according to the same $\gamma \in \Gamma$ in order to have consensus in global game. The next proposition formalises this intuition and provides a characterisation of the solution we are looking for.

Proposition 3.3.15 (Sink SCCs and Dimension of $\boldsymbol{\mu}$ Eigenspace) *Let H be the (row-stochastic) out-matrix of a digraph G and $(\mathcal{K}, \mathcal{E})$ its condensation graph. In these conditions, $|\boldsymbol{sink}(\mathcal{K})|$ is the dimension of the eigenspace associated to the eigenvalue $\lambda = 1$, i.e., the eigenspace of the stationary distributions of H. Moreover, $\mu(v) = 0$ iff $v \notin \boldsymbol{sink}(\mathcal{K})$ for any μ in the eigenspace of $\lambda = 1$.*

Proof. First, note that, because H is stochastic, $\lambda = 1$ is an eigenvalue, and therefore a stationary distribution exists. Let $\mathbf{sink}(\mathcal{K}) = \{K_1, \ldots, K_d\}$ and let H_i be the out-matrix of K_i, for $1 \leq i \leq d$. Note that no edge leaves each K_i. Then, the out-matrix of G can be written as

$$H = \left(\begin{array}{c:c:c:c} H_1 & 0 & 0 & 0 \\ \hdashline 0 & \ddots & 0 & 0 \\ \hdashline 0 & 0 & H_d & 0 \\ \hdashline * & * & * & M \end{array}\right) \tag{3.36}$$

Where $*$ represent any entries and M is substochastic matrix[xiv] (otherwise M would be the out-matrix of a sink SCC). Let $\widetilde{\mu}_i$ be the unique stationary distribution of H_i (recall Remark 3.3.10). It is not hard to see that, for each i, $1 \leq i \leq d$, the row vector μ_i defined as $\mu_i(v) := \widetilde{\mu}_i(v)$ if $v \in K_i$, and $\mu_i(v) := 0$ otherwise, is a stationary distribution of H. Thus, the dimension of the eigenspace of H associated to $\lambda = 1$ is greater than or equal to d.

We finally need to show that, if μ is a stationary distribution of H, then $\mu(v) = 0$ for $v \notin K_i$, for all i. Let μ be a stationary distribution of H. Let the $\widetilde{\mu}$ be vector formed by the last coordinates of μ, i.e., formed by the coordinates associated to $v \notin K_i$, for all i. Because values above M in Equation 3.36 are all 0, if $\widetilde{\mu}$ is non-zero, it would be a stationary distribution of M. However, by Perron and Frobenius Theorem (described in [12, Page 137]), we have that M, for being substochastic, does not admit a stationary distribution. Therefore $\widetilde{\mu}$ is the zero vector, and thus the dimension of the eigenspace of H associated to $\lambda = 1$ is d. ■

The proposition above is useful if we want to standardise our vector μ for a weakly connected graph G, with out-matix H. Each SCC $K_j \in \mathcal{K}$ of G can be seen as a induced subgraph of G and thus will have a stationary distribution normalised according to Definition 3.3.8. Then, we will have a sequence of μ_j, one for each K_j, such that their entries are the influence of each node v if $v \in K_j$ and null otherwise. We then define a standard stationary distribution μ of H as

$$\mu = \mu_1 + \cdots + \mu_{|\mathbf{sink}(\mathcal{K})|} \tag{3.37}$$

Note that μ is indeed such that $\mu H = \mu$ because $\{\mu_j\}_j$ form an eigenspace.

We finally address Question **C6** by presenting the following theorem.

Theorem 3.3.16 (Probability of Consensus in Digraphs) *Let $(\vec{\mathcal{F}}, S)$ on G be a consensus game, and let $\gamma \in \Gamma$ be a winning configuration. Let also $(\mathcal{K}, (E))$ be the condensation graph of G. Then the probability of consensus in γ given an initial configuration S_0 is given by*

$$Pr(S_\tau = \gamma \mid S_0) = \prod_{K \in \mathcal{K}} \prod_{i=1}^{k(K)} \Theta_i^\gamma. \tag{3.38}$$

[xiv] A substochastic matrix is such that its rows sum to at most 1, with at least one row adding up to a value strictly less than 1.

Proof. We combine Lemma 3.3.12 with Proposition 3.3.15. For each SCC K_j of G we apply Lemma 3.3.12 to get a stationary distribution and extending it to μ_j **for the entire graph** G by having zeros in all coordinates $\mu_j(v)$ when $v \notin K_j$. Because all SCC act independently, we need all of them to converge to the same γ in order to reach consensus. ∎

Remark 3.3.17. Note that the existence of state-loops on a given sink SCC does not necessarily imply the existence of state-loops on the entire graph G. In fact, not even the presence of state-loops in all sink SCCs is enough to characterise a global state-loop. The final necessary condition depends on how the edges of G connect all nodes in the graph to the SCCs. To be more precise, in order to always achieve a state-loop configuration in losing games, our graph G must be of the following form: all nodes v in non-sink SCCs must be such that all paths starting from v reach only one sink SCC K, and the length of all paths from v to any reachable $w \in K$ must be equivalent modulo $k(K)$. The size of the global loop will be equal to the minimum common multiple of the set $\{k(K)\}_{K\in\mathcal{K}}$.

Solution to Problem 4. Probabilities of convergence are given by

$$Pr(\bullet) = Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{12}{25} = 0.48. \tag{3.39}$$

$$Pr(\bullet) = Pr(S_\tau = \gamma_{\text{red}} \mid S_0) = \frac{2}{25} = 0.08. \tag{3.40}$$

The probability for this game not reaching consensus is 0.44. For a more detailed solution for this problem, see Example 4.4.11.

3.3.3 Another Small Generalisation

As in Section 3.2.4, we have assumed, for simplicity, that our nodes treat their neighbours equally. The results and the proof remain essentially the same if we require only that the probabilities of copying neighbours sum to one. Note that self loops are accepted, and thus the model allows nodes to have a positive probability of keeping their current colour. Thus, all theorems (and their proofs) remain the same if we replace "row-normalised out-matrix H", in Definition 3.3.2 by "row stochastic adjacent matrix H".

3.4 Summary of Results

In this chapter, we analysed synchronous generalised consensus games on graphs. In particular, we were interested in the probability that these games fail to reach consensus. These failures were characterised by state-loops in strongly connected components of G. Here we revisit the sets of questions raised in the beginning of sections and present their solutions.

B1: Losing configurations in undirected graphs have a positive probability of being achieved for at least one initial configuration if and only if G is bipartite.

B2: Theorem 3.2.26 gives us the probabilities of goal configuration $\gamma \in \Gamma$ to be achieved. Results are also in Table 3.1.

B3: Table 3.1 summarises all upper and lower bounds found for the expected duration of games on undirected bipartite graphs.

B4: Our analysis also concludes that any configuration other than consensus might lead to an infinite loop in bipartite graphs. That reinforces our definition that winning games are only the ones that already reached consensus.[xv]

C1: Examples of graphs that admit infinite loops of configurations of size 3 are tripartite digraphs. (see Figure 3.12).

C2: Lemma 3.3.7 gives us the characterisation that only games of the form $(\vec{\mathcal{F}}_k, S_0)$ might enter a state-loop of size k.

C3: For games of the form $(\vec{\mathcal{F}}_k, S_0)$, we can define a split function that takes configuration S_0 as input and outputs k configurations, each one formed by copying the colours of a different partition in S_0 and colouring all the other partitions according to a common given goal configuration $\gamma \in \Gamma$.

C4: Lemma 3.3.12 gives us the probability that a game $(\vec{\mathcal{F}}_k, S_0)$ reaches consensus in a given winning configuration $\gamma \in \Gamma$.

C5: Games on weakly connected digraphs might admit losing games with no state-loops on G (see Remark 3.3.17), although loops might be present in sink SCCs of G.

[xv] Note that the same is not true for losing games if we consider more than 2 colours and have no colour present in both partitions. In these situations, the game has no chance of being a winning game at a point in which partitions might not yet be all monochromatic.

C6: Theorem 3.3.16 presents the probability of a given winning configuration $\gamma \in \Gamma$ to be achieved in a game $(\vec{\mathcal{F}}, S_0)$ on a weakly connected graph G.

Chapter 4

Team Persuasion Games

4.1 Introduction and Motivation

Argument-based persuasion dialogues provide an effective mechanism for agents to communicate their beliefs and reasoning in order to convince other agents of some central topic argument [60]. In complex environments, persuasion is a distributed process. To determine the acceptability of claims, a sophisticated agent or audience should consider multiple, possibly conflicting, sources of information that can have some level of agent-hood. In this chapter, we consider teams of agents that work together in order to convince some audience of a topic argument. While strategic considerations have been investigated for one-to-one persuasion (e.g. [73]), and for one-to-many persuasion (e.g. [36]), the act of persuading as a team is a largely unexplored problem.

Consider a political referendum, where two campaigns seek to persuade the general public of whether or not they should vote for or against an important proposition. Each campaign consists of separate agents, where each agent is an expert in a single argument. For example, an environmentalist might argue how a favourable outcome in the referendum would reduce air pollution. Each agent can assert its argument to the public, and each agent is aware of counterarguments that other agents can make. However, no agent can completely grasp all aspects of the campaign, for example the environmentalist may be ignorant of relevant economic issues. If the agent thinks there are no counterarguments to its argument, then it should keep asserting its argument, as it is beneficial for its team. While each agent wishes to further their team's persuasion goal, they do not want to risk having their argument publicly defeated by counterarguments.

From this example, we consider a team of agents to have three key properties that differentiate them from an individual agent when persuading. Firstly, each agent may have localised knowledge which is inaccessible and non-communicable to other agents in the same team. Secondly, agents may not be wholly benevolent, potentially acting in their own interest before that of their team; reconciling this conflict between individual and team goals makes strategising more complex. Thirdly, there is no omniscient or authoritative agent able to determine the actions of the other agents in the team, meaning each agent must act independently, making the problem a distributed one. This problem is distinct from that of an individual persuader, and therefore requires a different approach to model the outcomes of persuasion.

We approach the problem of modelling team persuasion by exploring a particular team persuasion game, in which two opposing teams attempt to convince an audience of whether some central issue, termed the *topic*, is acceptable or not. For simplicity, we assume that each agent in a team is individually responsible for one argument in the domain, being strongly associated to that particular argument in audience's perception. As such, each agent must independently decide whether to actively assert its argument to the audience, or to hold back from asserting its argument. The persuasion game proceeds in rounds, where in each round an agent decides whether to assert its argument. An agent can decide to stop asserting its argument even if in previous rounds they had asserted it. Teams aim to reach a state in which the topic is acceptable or unacceptable according to the audience (depending on whether the agent is defending or attacking the topic), and in which no individual agent will change its decision of whether to assert (reinforce) its argument; in such a state the topic is guaranteed to retain its (un)acceptability indefinitely. When deciding whether to assert its argument, an agent takes into account whether the other agents are currently asserting their arguments. It aims to have a positive effect on its team's persuasion goal, but may also wish to avoid having its own argument publicly defeated (since this may, for example, negatively affect their public standing or reputation). When deciding whether to assert its argument, the agent must therefore balance the potential positive effect of this on its team's persuasion goal with the risk of its own argument being publicly defeated.

The audience determines whether they find the topic argument acceptable in a particular round by considering the set of arguments that are currently asserted. Note that the audience has no knowledge of which arguments were asserted in previous rounds; we consider the audience to be memoryless, only considering the arguments that are asserted in the current round.

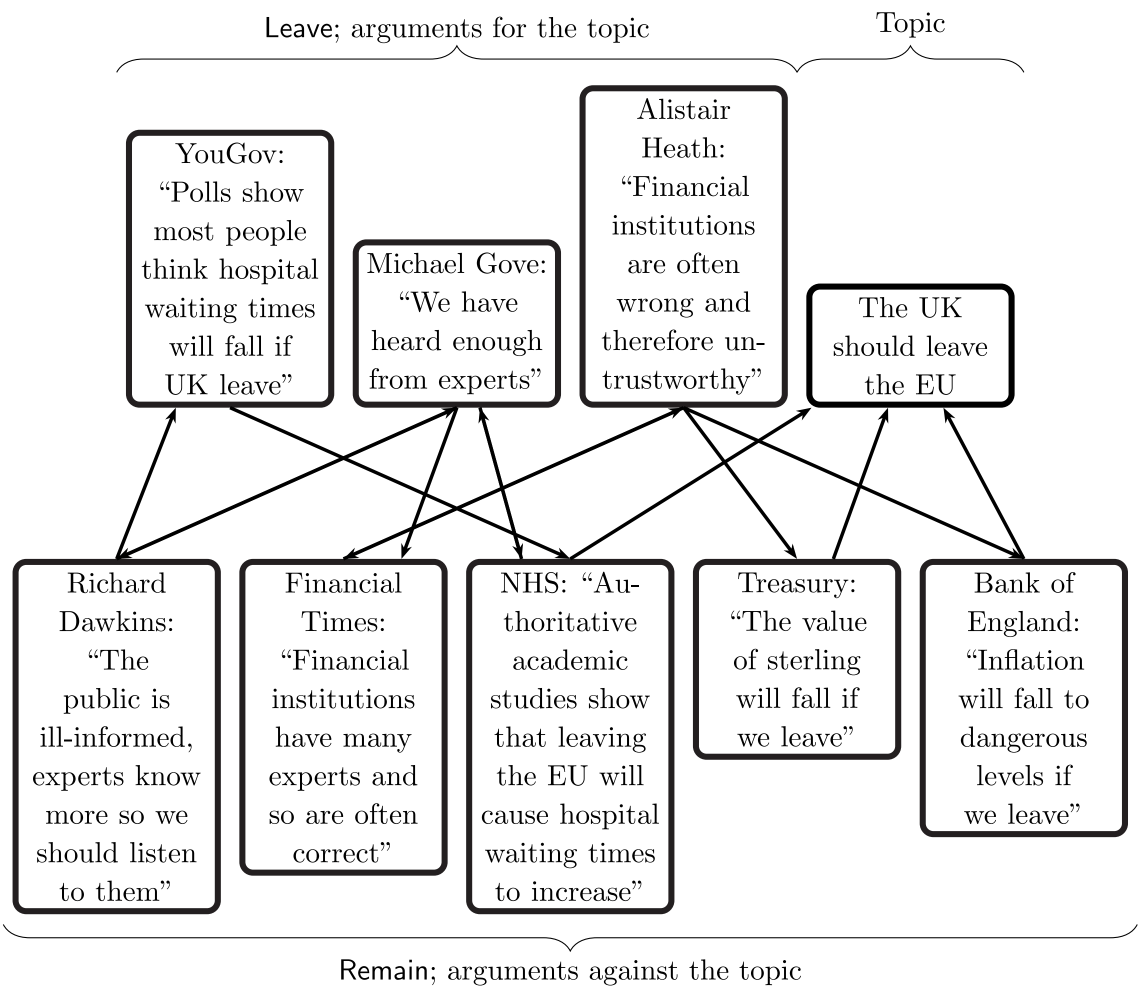


Figure 4.1: An instantiated example of a bipartite argumentation framework. A Possible Debate Prior to the 2016 Vote for Britain to Leave the European Union.

For example, consider the arguments in Figure 4.1, in which the directed edges represent conflict between arguments. The topic argument in this example is that the United Kingdom should leave the European Union, with three arguments defending the topic and five arguments attacking the topic (some indirectly). Each argument is controlled by a particular individual or institution. The agents are organised into two teams, those defending the topic (the Leave campaign), and those attacking the topic (the Remain campaign). Consider the argument that might be asserted by the Treasury: the Treasury is motivated to assert their argument as it directly attacks the topic argument (which they are seeking to dissuade the audience of). If they are aware of the argument possibly asserted by Alistair Heath, they may decide not to assert their own argument to avoid the risk of being publicly defeated. The public decides whether leaving the European Union is acceptable depending on which arguments are currently being asserted.

The perceived acceptability of the topic of the dialogue can also be of interest to the audience, who are not themselves interlocutors, but are observing the course of the persuasion dialogue. Previous work has considered how the values of an audience can determine how the interlocutors should argue in order to be persuasive (e.g. [5]). Though unable to assert arguments themselves, we consider how external agents may be able to influence the dialogue towards a preferred outcome through bribing the interlocutors. We use the term *bribery* as the offer of an incentive to an interlocutor so that the interlocutor behaves in a way that increases the likelihood that the dialogue will result in that audience member's preferred outcome.

Here, potential bribers must balance the increase in utility, which we will formalise as the increase in the probability for a favourable outcome, against the loss of utility through the cost of the bribe. This raises strategic questions for them, for example, which interlocutors should be bribed? How should their behaviour be changed? How much incentive should be offered? We begin by analysing the decisions to be made when there is only one briber, and then expand to a two-briber scenario.

The contribution of this chapter is the application of Flag Coordination Games to model public debates of this form. We are also introducing the concept of bribery in Flag Coordination Games. We answer the following:

D1 How do we formalise the situation where one team has *definitively won*? We define such a situation to be a state where agents that are asserting their arguments will continue to do so, and agents not asserting their arguments will never do so.

D2 What is the probability that a particular team (e.g. the Remain Campaign) has definitively won? We prove an expression for this probability, given the initially asserted arguments and the attacks between them.

D3 Single briber: We introduce an external agent - the briber - who at a given point in the dialogue can sway any interlocutor to start or stop asserting their argument. Assuming that the briber acts to maximise their expected utility, *which interlocutor should be bribed, and how much should the briber be willing to pay?*

D4 Two bribers: We now consider two bribers who at a given point in the dialogue simultaneously make a decision about which interlocutors they will bribe. *How should each briber amend the answer to the above question if there is another such briber?*

In Section 4.2 we provide the necessary Argumentation Theory background. In Section 4.3 we define a team persuasion game on a bipartite abstract argumentation framework [25], which is a special case of a Flag Coordination Game (in digraphs) seen in Chapter 3. In Section 4.4, we use our framework and results from Chapter 3 to answer Questions **D1** and **D2**. Finally, in Section 4.5, we answers Questions **D3** and **D4**. We discuss related work in Section 4.6, and conclude in Section 4.7.

4.2 Argumentation Theory

In this section we present our model of team persuasion games. We begin by briefly reviewing the relevant aspects of abstract argumentation [25].

Definition 4.2.1 *An* ***argumentation framework*** *is a directed graph (digraph)* $AF := \langle A, R\rangle$ *where* A *is the set of arguments and* $R \subseteq A \times A$ *is the attack relation, where* $(a, b) \in R$ *denotes that the argument* a *attacks the argument* b.

Figure 4.1 is an example argumentation framework. We will only consider *finite*, non-empty argumentation frameworks, i.e. where $A \neq \varnothing$ is finite. Given an argumentation framework, we can determine which sets of arguments (*extensions*) are justified given the attacks [25]. There are many ways (*semantics*) to do this, each based on different intuitions of justification. We do not assume a specific semantics in this chapter, only that all agents and the audience use the same semantics.

Definition 4.2.2 *Let AF be an argumentation framework. The set* $\mathrm{ACC}(AF) \subseteq A$ *is* ***the set of acceptable arguments of*** *AF, with respect to some argumentation semantics under credulous or sceptical inference. An argument a is said to be* ***acceptable*** *with respect to AF iff $a \in \mathrm{ACC}(AF)$.*

We now define a refinement of the concept of neighbourhood for directed graphs, taking in account the direction of the attacks.

Definition 4.2.3 *Let $AF = (A, R)$ be an argumentation framework and $a \in A$. Define* ***the set of arguments attacked by*** *a as $a^+ := \{b \in A \mid (a, b) \in R\}$, and* ***the set of arguments attacking*** *a as $a^- := \{b \in A \mid (b, a) \in R\}$.*

4.3 Team Persuasion Games

We model team persuasion as an instance of a Flag Coordination Game over an argumentation framework. As we have seen before in this dissertation, such models have been studied in the context of the adoption of new technology standards, voting and achieving consensus, and also in the context of failure of consensus in synchronous protocols. But what exactly are we looking for and how can we frame it as a Flag Coordination Game? We will gradually define terms in what we will call a *Team Persuasion Game*. More formally, $\mathcal{F}_{\mathrm{TP}} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$. For now, we say that $G = AF$.

Definition 4.3.1 *A* ***team persuasion framework*** *is a tuple given by $\mathcal{F}_{TP} = \langle AF, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$. Let $AF = (A, R)$ be an argumentation framework, where the nodes represent arguments, each owned by distinct agents.*[i] *Let $\phi : A \to \mathbb{P}(A)$ be* ***the visibility function****, i.e. $\phi(a) \subseteq A$ is the set of arguments that a can see. Let $X := \{on, off, topic\}$ denote* ***the set of opinions****, or* ***colours*** *in this game. Let $\boldsymbol{t} \in A$ be a distinguished argument called* ***the topic (argument)****. Define $\beta : A \times T \to \mathbb{P}(X)$ as the function that associates a set of possible colours, or flags, to an argument a at a given time. Unless otherwise state, we fix β over time for a given a. We then define $\beta(\boldsymbol{t}) := \{topic\}$ and $(\forall a \in A \setminus \{t\})\, \beta(a) \in \{on, off\}$.*

Let $\mathcal{S} := X^A$ be the space of functions that assigns a state to each argument, which defines a ***configuration****. Let $\Gamma \subseteq \mathcal{S}$ be* ***the set of goal states****. For $a \in A$*

[i] As each *argument* is owned by a distinct *agent*, we use the terms interchangeably.

let α_a be ***the decision algorithm of agent*** *a, that takes as input T, β, ψ, Γ, and ϕ and outputs $S(a) \in X$, for $S \in \mathcal{S}$. We define $\mathcal{A}$ as the set of algorithms for all $a \in A$.*

The team persuasion framework is such that each agent asserts a single argument, which can attack and be attacked by other asserted arguments, so it is isomorphic to an argument framework. Each of the agents can assert their argument (turning it *on*) or not assert their argument (turning it *off*). The topic is a special argument that is labelled *topic* throughout the duration of the game.

Definition 4.3.2 (Team Persuasion Game) *Let $\mathcal{F}_{TP}$ denote a team persuasion framework. Let T be a discrete time set. Let $\{S\}_{t\in T}$, be a random variable that describe the configurations of this game over time. We call S_0* ***the initial configuration****, and S_t is* ***the t^{th} configuration****. The update rule is such that for all $a \in A \setminus \{\boldsymbol{t}\}$, $S_{t+1}(a) \in X$ is the output of α_a given $S_t(b) \in X$ for all $b \in \phi(a)$ and possibly $\beta(a)$. Further, $(\forall t \in T)\, S_t(\boldsymbol{t}) :=$ topic. Arguments make their decision at the end of round t and change at the start of round $t+1$.* ***A team persuasion game with initial configuration*** *S_0 is the tuple $(\mathcal{F}_{TP}, S_0)$.*

Initially, the agents start in some initial configuration defined by whether each agent asserts its argument. In each subsequent round, the agents decide using their own decision procedure whether to assert or stop asserting their argument in the next round, given the actions of other agents they see.

Both teams are presenting their arguments to an audience who are assumed to be memoryless across rounds and can only see the topic and the arguments that are being currently presented. This prompts the following definition.

Definition 4.3.3 *Given a Team Persuasion Game,* ***the set of arguments that are on in round*** *t is $A_t^{on} := \{a \in A \mid S_t(a) = on\} \cup \{\boldsymbol{t}\}$. The* ***induced argument framework*** *is $AF_t^{on} := \langle A_t^{on}, R_t^{on}\rangle$, where $R_t^{on} := R \cap [A_t^{on} \times A_t^{on}]$.*

The audience will therefore see a sequence of argument frameworks $(AF_t^{\mathrm{on}})_{t \in t}$ as the teams debate each other about the topic. The audience can determine which team is winning based on whether the topic is justified in a given round.

Definition 4.3.4 *In a given round $t \in t$ of a team persuasion game, we say that* ***the team of defenders are winning*** *iff $\boldsymbol{t} \in \textsc{Acc}\,(AF_t^{on})$ iff* ***the team of attackers are not winning****.*

In each round the acceptability of the topic may change, and hence the winner can change. We are interested in definitively winning states, as defined in **D1** in Section 4.1. We explore the existence of such states in Section 4.4.

Since we are modelling the arguments between two teams, each trying to persuade or dissuade an audience of the topic, we specialise to bipartite argumentation frameworks because no agent should attack an argument of another agent in its own team. Further, the framework is weakly connected because all arguments asserted are relevant to the debate. Further, we assume that every argument has a counterargument, and that the topic is not capable of defending itself, so it does not directly attack any argument.

Definition 4.3.5 *Team persuasion frameworks* $\mathcal{F}_{TP} = \langle AF, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$ *have an underlying argument framework* $AF = (A, R)$ *that is bipartite and weakly connected, with the requirements that* $(\forall a \in A)\, a^- \neq \varnothing$ *and* $\boldsymbol{t}^+ = \varnothing$.[ii] *A team's goal is to make the topic acceptable or unacceptable to the audience, depending on whether they are the team for or against the topic, respectively. The teams each form their own partition of the bipartite* AF*, which we denote* $A = P_{for} \cup P_{ag}$ *such that the two* $\subseteq$*-maximal independent sets are* $P_{for} \cup \{\boldsymbol{t}\}$ *and* P_{ag}*, where partition* P_{for} *is the team that is* ***for*** *the topic* $\mathbf{t}$ *and* P_{ag} *is the team that is* ***against*** $\boldsymbol{t}$.[iii] *As a digraph, we assume that this AF is weakly connected, where all arguments are attacked by some other argument, and* $\boldsymbol{t}$ *attacks no argument.*

More formally, we have that the ***set of goal states*** *is* $\Gamma := \{\gamma_{for}, \gamma_{ag}\}$*, where* $\gamma_{for}(P_{for} \setminus \{\boldsymbol{t}\}) = \{on\}$ *and* $\gamma_{for}(P_{ag}) = \{off\}$*, and* $\gamma_{ag}(P_{for} \setminus \{\boldsymbol{t}\}) = \{off\}$ *and* $\gamma_{ag}(P_{ag}) = \{on\}$.[iv]

Intuitively, these requirements formalise the idea that all arguments that can be put forward can be criticised, and are relevant to the topic. As each argument is being asserted by a distinct agent who is an expert in that argument, we will use the terms *agent* and *argument* interchangeably.

The goal states indicate that each team has the goal of unilaterally asserting their arguments and making the opposing team unilaterally withdraw their arguments. See Figure 4.2 for an example of γ_{for}, and Figure 4.3 for an example of γ_{ag}. In our figures, white (resp. black) nodes are arguments that are *on* (resp. *off*).

[ii] Recall that for an $\text{AF}\langle A, R\rangle$ where $a, \mathbf{t} \in A$, $a^- := \{b \in A \mid (b,a) \in R\}$, and $\mathbf{t}^+ := \{b \in A \mid (\mathbf{t}, b) \in R\}$.

[iii] We assume that $P_{\text{for}}, P_{\text{ag}} \neq \varnothing$, so $A \cup \{\mathbf{t}\}$ has at least 3 arguments.

[iv] Recall that for a function $f : X \to Y$ and $A \subseteq X$ the *image set of* A *under* f is $f(A) := \{y \in Y \mid (\exists x \in A)\, f(x) = y\}$.

Note that arguments that are against the topic do not have to necessary attack $\mathbf{t}$ directly. Moreover, since we do not require AF to be strongly connect, there may not be a path from every argument to the topic. In case there is one, the parity of its length will be determined by the partition the given agent is: path of even length for $a \in P_{\text{for}}$ and odd length for $a \in P_{\text{ag}}$.

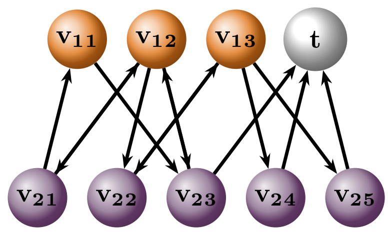


Figure 4.2: The defenders' goal state γ_{for}; all defenders are asserting their argument.

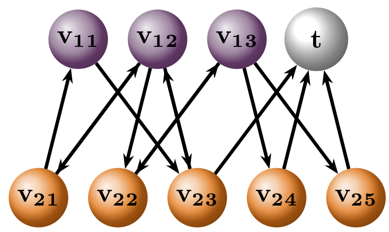


Figure 4.3: The attackers' goal state γ_{ag}; all attackers are asserting their argument.

4.3.1 The Scheduler and Agent Visibility

It is somehow unrealistic to suppose that agents can only see their neighbours, as we did for problems studied in Chapter 3. If a member of the audience may see the entire argumentation framework, why cannot debaters? We will then assume that agents participating in the debate have visibility given by $\phi(a, t) = A$, for all $a \in A \setminus \{\mathbf{t}\}$ and all $t \in T$. Which does not mean that they will equally consider all nodes when making their own decision of being *off* or *on*. In next section (Section 4.3.2), we will better detail how agents can differently use the various layers of their visibility.

The scheduler of a team persuasion game can be either such that nodes act synchronously or such that they act asynchronously.

(i) The scheduler σ is such that agents act synchronously. This models situations in which agents are expected to act somehow together (e.g. from one day to the next). Formally, for all $t \in T$, we have $\sigma(t) = A$.

(ii) The scheduler σ is such that agents act asynchronously chosen by the scheduler uniformly at random. This models situations of more dynamic public debate, for example, in which agents revise their state independently of the other agents in the network. Formally, for each $a \in A$ and for all $t \in T$, we have $\sigma(t) = a$, with probability $|A \setminus \{\mathbf{t}\}|^{-1}$.

In the sections that follow, we are going to provide solutions for both cases. We can already observe that losing configurations are only possible with a synchronous scheduler.

4.3.2 The Agents' Decision Algorithm

Although agents can see the entire argumentation framework AF, it is reasonable to expected that an agent might give higher importance to an immediate attacker rather than to an indirect distant one. They also desire to make the topic acceptable/unacceptable to the audience (the goal of the team), at the same time as not having their argument publicly defeated (the goal of the individual). An individual does not want to have its argument publicly defeated (that is, its argument is asserted but is not considered acceptable by the audience in the current round), as it is somehow a challenge to the agent's authority. An agent can estimate how likely it is that their argument will be publicly defeated in the short- or long-run by considering different levels:

Level 1: Agents take into account solely their (immediate) attackers, i.e., the set $a^- := \mathbf{level}_1(a^-)$, with an arbitrary distribution of weights on each attacking argument. We denote the sum of weights on Level 1 by $\mathbf{w}_1$.

Level 2: Agents take into account the set of attackers of their attackers, i.e., the set $(a^-)^- := \mathbf{level}_2(a^-)$. We denote the sum of weights on Level 2 by $\mathbf{w}_2$.

$\vdots$

Level L: Agents consider the set $\mathbf{level}_L(a^-) := \mathbf{level}_2(a^-)^-$. We denote the sum of weights on Level L by $\mathbf{w}_L$.

Definition 4.3.6 *Given an agent $a \in A \setminus \{\boldsymbol{t}\}$, let L be the highest integer for which $\boldsymbol{w}_L > 0$, in this conditions we say that a is a L-**agent**. Moreover, we denote the weight agent a assigns to agent b as $\boldsymbol{w}(a \to b)$.*

Although different agents in the same game may have different assignment of weights to other agents, as well as consider different levels of neighbourhood, we assume, for a given agent a, weights maintain unchanged for the duration of the game.

Although weights may be assigned freely by each agent, it is expected that the weight on each agent $b \in \mathbf{level}_{L-1}(a^-)$ takes into account the weight of their attackers $(b)^- \subset \mathbf{level}_L(a^-)$.

The state of agents in even and odd levels transmit opposite information to agent a. For odd levels, agent a can estimate how likely it is that their own argument is successful by how many attacking arguments the agent could see are being asserted: the more attackers that are asserted, the more likely one of the attacks will be successful (either directly or indirectly), and therefore the higher the chance its argument is defeated. On the other hand, for even levels, agent a can estimate how likely it is that their own argument is successful by how many attacking arguments the agent could see are being asserted: the more arguments that are asserted, the more likely one of the attacks will be successful in defeating an argument from the other team, and therefore the higher the chance a is accepted.

- **Altruistic**: An agent which is only motivated by the team goal of making the topic (un)acceptable would always assert its argument a, regardless of the state of the arguments up to $\mathbf{level}_L(a^-)$. We call such selfless agents *altruistic.*
- **Timid**: An agent which is only motivated by its individual goal of not having its argument being publicly defeated would never assert its argument, regardless of which arguments up to $\mathbf{level}_L(a^-)$ are being asserted. If the agent never asserts its argument, it can never be defeated, and therefore will always achieve its individual goal.
- **Balanced**: An agent motivated by both factors must find a way to balance these two goals. Such an agent is certain to assert its argument when none of its attackers in up to $\mathbf{level}_L(a^-)$ are asserted and all of its defenders in up to $\mathbf{level}_L(a^-)$ are, because the chance of a successful defeat is minimal. Similarly, the agent is least likely to assert when all of its attackers in up to $\mathbf{level}_L(a^-)$ are asserted and all of its defenders are not, because the chance of successful defeat is maximised.

In this chapter, we will consider our analysis based on balanced agents, leaving the rest for future work. We define the probability of the agent, based on its weights given across all levels, not asserting its argument when all of its attackers are *on* and defenders are *off* as 1, and conversely the probability of the agent not asserting its argument when all of its attackers are *off* and defenders as 0. We provide a formal definition as follows.

Definition 4.3.7 *Let $\mathcal{F}_{TP}$ be a team persuasion framework on an argument framework AF as defined in Definition 4.3.5. An agent $a \in A \setminus \{\boldsymbol{t}\}$ is* ***balanced*** *iff* α_a

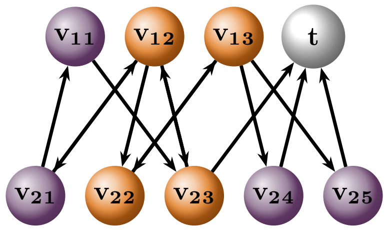


Figure 4.4: An Initial Configuration $(\mathcal{F}_{\mathrm{TP}}, S_0)$ for the example in Figure 4.1.

(Definition 4.3.1) is defined as follows. Denote $\boldsymbol{w}_l(\mathrm{on})$ and $\boldsymbol{w}_l(\mathrm{off})$ as the sum of the weights of agents in Level l that are currently on *and* off*, respectively. For $t \in T$, α_a outputs $S_{t+1}(a) = off$ with conditional probability*

$$Pr\left(S_{t+1}(a) = off \mid S_t\right) = \frac{\sum_{l\ odd}^{L} \boldsymbol{w}_l(\mathrm{on}) + \sum_{l\ even}^{L} \boldsymbol{w}_l(\mathrm{off})}{\sum_{l=1}^{L} \boldsymbol{w}_l} \in [0,1]. \tag{4.1}$$

Further, α_a outputs $S_{t+1}(a) = on$ given S_t with probability $1 - Pr\left(S_{t+1}(a) = off \mid S_t\right)$. We will assume that for all $a \in A \setminus \{\boldsymbol{t}\}$, a is balanced.

Example 4.3.8. Consider Figure 4.4, which represents the situation in Figure 4.1 as a team persuasion framework with the initial configuration where the *on* arguments are v_{12}, v_{13}, v_{22}, and v_{23}, with the rest of the arguments being *off*. Consider the argument v_{23}. Consider that it is a 1-agent, with $\mathbf{w}_1 = 1$ and uniformly distributed among its immediate neighbours (set v_{23}^-). It is attacked by v_{11} and v_{12}, which are respectively *off* and *on*. Therefore, the probability of v_{23} remaining *on* in the next round is $\frac{1}{2}$.

4.4 Reaching State-Stable Configurations

From the setup described in Section 4.3, we can now more formally define Questions **D1** and **D2** as follows.

D1' Are there any states of the arguments (*on* or *off*) in which no agent is going to change their state in any future round according to α_a as defined in Equation 4.1? We call such a state a *state-stable configuration.*[v]

D2' What is the probability of a particular team winning, i.e. achieving a state-stable configuration, where the topic is either acceptable or unacceptable?

[v]This is to avoid confusion with the notion of *stable semantics* [25].

4.4.1 State-Stable Configurations

We now answer Question **D1**, which concerns state-stable configurations.

Definition 4.4.1 *A **state-stable configuration** is a function $s \in \mathcal{S}$ such that, if attained at round $t \in T$ of the team persuasion game following Equation 4.1, will also be the state of the game in all subsequent rounds.*

This formalises the intuition that no agent is going to change their state in any future round once a state-stable configuration is reached.

We now identify the state-stable configurations which are desirable for each team. A state-stable configuration is considered a winning state by a team only if the topic has the desired acceptability in that state.

Proposition 4.4.2 *Given the setup of Section 4.3, the two goal states, γ_{for} and γ_{ag} (Definition 4.3.5) are the only state-stable configurations.*

Proof. To show that γ_{for} is a state stable configuration, notice that in round $t \in T$, if γ_{for} is attained, then for $a \in P_{\text{for}} \setminus \{\mathbf{t}\}$, the probability (Equation 4.1) a will be off in round $t+1$ is zero, because $a^- \subseteq P_{\text{ag}}$ and all attackers of a are off. Therefore, a will still be on in round $t+1$. Similarly, we can show that the probability of being off for all $b \in v_2$ in round $t+1$ is one. Therefore, in round $t+1$, the state is still γ_{for}. A similar argument to this proves that if γ_{ag} is attained in round t, then it will also be the state for round $t+1$. By induction over i, γ_{for} and γ_{ag} satisfy Definition 4.4.1.

We now show that both γ_{for} and γ_{ag} are the only state stable configurations. Assuming the contrary. Then, we have a configuration different from γ_{for} and γ_{ag} in which no argument has a positive probability of changing their state. In this case, we would have two nodes, say v_{11} and v_{12}, in the same partition, say P_{for}, that have different colours (otherwise we have γ_{for} and γ_{ag}). Since G is weakly connected, there is a path that ignores edges' directions from v_{11} to v_{12}. This path has even length and, therefore, since v_{11} to v_{12} are different, there must be at least two consecutive nodes in this path with the same colour. One it attacking the other, therefore, the attacked one has a positive probability of changing their colour. We have a contradiction. Thus γ_{for} and γ_{ag} must be the only state-stable configurations in a bipartite AF. ■

4.4.2 Probabilities for State-Stable Configurations

We now answer Question **D2** for synchronous and asynchronous games. We first translate our team persuasion game into a *consensus game* . Recall that, in a consensus game, the update is such that in round $t+1$, every digraph node *a copies* the colour of a randomly (uniformly or not) sampled neighbour in a^+, rather than adopting the opposite colour as in Equation 4.1.

Note that this procedure is partially necessary but partially not. Although we do need to reverse the edges, and take in account weights assigned from one argument to another, in order to frame this as a generalised consensus game, we did not need to transform an anti-consensus into a consensus game. The reason is because both are generalised consensus games and our analysis in Chapter 3 takes them all into account. The reason why this change will be made is so that visualisation becomes easier throughout the remainder of this chapter.

4.4.2.1 The translation to a consensus game

The translation from proper colouring a bipartite graph into consensus is straight forwards. The procedure is as described now in detail.

We consider the finite, weakly connected, bipartite digraph $G = (V, E)$ which is the induced subgraph of $\langle A, R\rangle$ with nodes $:= A \setminus \{\mathbf{t}\}$. For each configuration $s : A \setminus \{\mathbf{t}\} \to \mathcal{S}$, where $\mathcal{S} = \{\text{on}, \text{off}\}$ we define a *colouring function* $\overline{s} : V \to X'$, where $X' := \{0, 1\}$ such that

$$\overline{s}(a) := 1 \text{ if } \left[(a \in P_{\text{for}} \text{ and } s(a) = \text{on}) \text{ or } (a \in P_{\text{ag}} \text{ and } s(a) = \text{off})\right]. \tag{4.2}$$

$$\overline{s}(a) := 0 \text{ if } \left[(a \in P_{\text{for}} \text{ and } s(a) = \text{off}) \text{ or } (a \in P_{\text{ag}} \text{ and } s(a) = \text{on})\right]. \tag{4.3}$$

Usually, we intuitively associate the colour 1 with the state on and similarly, 0 with off, but notice how this association is swapped for $a \in P_{\text{ag}}$. Thus, the correspondence $s \mapsto \overline{s}$ is well-defined and bijective.

Note that, as before, we use notation $\overline{s}$ for a possible colouring when it is not part of a random process, whereas $\overline{S}$, although also a colouring drawn from the exact same set $\mathcal{S}$, will be used when referring to a configuration indexed by time. That way we can denote situations such as "*let* $\overline{s} = (0, 0, 1, 1, 0, 1, 0, 1)$ *be a colouring of a graph* G*. Consider a game that starts with that configuration, i.e.,* $\overline{S}_0 = \overline{s}$."

Example 4.4.3. Consider the digraph in Figure 4.4. Given this initial configuration S_0 such that $S_0(v_{11}) = \text{off}$, $S_0(v_{12}) = \text{on}$... etc. (see Example 4.3.8), we get a corresponding $\overline{S}_0$ where $\overline{S}_0(\{v_{12}, v_{13}, v_{21}, v_{24}, v_{25}\}) = \{1\}$ and $\overline{S}_0(\{v_{11}, v_{22}, v_{23}\}) = \{0\}$, by Footnote iv. If we arrange $V = \{v_{11}, \ldots, v_{13}, v_{21}, \ldots, v_{25}\}$, we can represent $\overline{S}_0$ as the boolean vector $(0, 1, 1, 1, 0, 0, 1, 1)$.

Given this translation of a proper colouring game to a consensus game, how can we translate Equation 4.1 to give the one-step updating process for S_t to S_{t+1}? The intuition is that starting with the adjacency matrix M of G, which contains information on which node points to (attacks) which other node, we take the transpose M^T, which contains information on which node is pointed at (attacked by) which other node. We then row normalise M^T to capture the consensus game where each node a copies the colour of a randomly and (not necessarily) uniformly sampled neighbour of a^-. Recall the more formal Definition 2.4.4, in Chapter 2.[vi]

Definition 4.4.4 (Weighted Adjacency Matrix) *Let $(\mathcal{F}_{TP}, \overline{S}_0)$ be the consensus version of a team persuasion game as in Definition 4.3.2 with balanced agents on a bipartite $AF = (A, R)$ with initial colouring $\overline{S}_0$. Recall that $\boldsymbol{w}(a \to b)$ is the weight assigned by a to b. We are going to define the matrix H_{TP} as the weighted adjacency matrix of game $(\mathcal{F}_{TP}, \overline{S}_0)$ as follows: for every pair of arguments $a, b)$, we have*

$$(H_{TP})_{ab} = \frac{\boldsymbol{w}(a \to b)}{\sum_{a' \in A} \boldsymbol{w}(a \to a')} \tag{4.4}$$

Note that H_{TP} is always a stochastic matrix. We denote G_{TP} the digraph associated to H_{TP}.

Note that matrix H_{TP} represent the probability that a chooses an argument b to copy its (consensus-version) state. We can now work with graph G_{TP} almost exactly the same as with digraphs in Theorem 3.3.16 presented and proven in Chapter 3. The only difference has to do with the position of the topic argument in the condensation graph. In order to do that, we recall the definition of the condensation (di)graph of a given graph (Definition 3.3.13) by an example[vii].

Example 4.4.5. The condensation of Figure 4.4 is Figure 4.5. The only source component is $\{v_{11}, v_{12}, v_{21}, v_{23}\}$. If we consider the consensus version of the same game, we have a condensation graph as in Figure 3.15 from Chapter 3.

[vi] In the context of team persuasion games, we write all nodes in P_{for} first and then the nodes in P_{ag}, as in Example 4.3.8.

[vii] Note that this example is essentially the same as the one provided in Problem 4, with just its edges reversed.

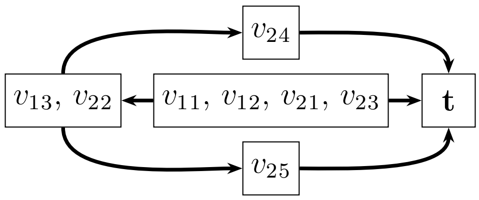


Figure 4.5: Condensation Graph of Figure 4.4, Showing Strongly Connected Components.

REMARK 4.4.6. Note that the condensation graph does not change when adding the weights of different layers of attackers to each of the arguments in the framework. That is because no path from a to b was created if it did not exist before.

4.4.2.2 Probabilities in Synchronous Games

The following theorem (which is just a slight generalisation of Theorem 3.3.16) answers Question **D2** with an analytic expression of the probability of a particular team winning for synchronous games. Intuitively, we first look at the condensation of a given bipartite AF (which is the same as the one of G_{TP} but with the edges reversed). Since source components (resp. sink components for G_{TP}) are not going to be influenced by any external argument, the probability of them reaching either one of the state-stable configurations is independent of the eventual state of the rest of the network. Thus, we need all source SCCs (resp. sink SCCs in G_{TP}) to converge to the same state-stable configuration, otherwise a global state-stable configuration will not be reached. Finally, in order to calculate the probability of either the defender or the attacker to win in each source SCC, we find each individual agents' influence on the network.

In sum, we are looking at the following differences when compared to our analysis in Chapter 3.

(i) Source SCCs in Team Persuasion Games play the role of Sink SCCs in generalised consensus games. That is because agents decide on their colours based on incoming edges, and not outgoing ones. However, if we use graph G_{TP}, we are back into considering only Sink SCCs.

(ii) Not all Source SCCs are important for the final consensus given that only the ones that lead to the topic are to be considered.

Definition 4.4.7 *Let $\mathcal{K} = \{\{\boldsymbol{t}\}, K_1, \ldots, K_m\}$ be the set of SCCs of AF (for some $m \in \mathbb{N}^+$), where $\{\boldsymbol{t}\}$ is the component that contains only the topic argument. We* *also define* $\textbf{\textit{source}}_{\mathcal{K}} \subseteq \mathcal{K}$ *as the set of source SCCs in the condensation of AF. Let* $\mathcal{K}_{\{t\}} \subseteq \textbf{\textit{source}}_{\mathcal{K}}$ *denote the set of SCCs for which there is a* $\mathcal{E}$*-path in the condensation of AF to* $\{\boldsymbol{t}\}$.

Proposition 4.4.8 *Let* $\left(\mathcal{F}_{TP}, \overline{S}_0\right)$ *be the consensus version of a team persuasion game as in Definition 4.3.2 with balanced agents on a bipartite* $AF = (A, R)$*. Then, this game admit losing configurations if and only if each strongly connected component* $K \in \mathcal{K}_{\{t\}}$*, the subgraph of* G_{TP} *induced by* K *is bipartite.*

Proof. The proof is immediate by observing that if G_{TP} is bipartite, then $k(G_{\mathrm{TP}}(K))$, i.e., the greatest common divisor of the length of all cycles in the subgraph of G_{TP} induced by each $K \in \mathcal{K}_{\{t\}}$, is strictly greater than 1. We then apply Lemma 3.3.7. ■

Remark 4.4.9. Note that G_{TP} is bipartite if and only if $\mathbf{w}_i = 0$ for i even and for every $a \in A$.

The following theorem generalises Theorem 3.3.16 in order to take into account only nodes that can directly or indirectly influence the topic argument.

Theorem 4.4.10 (Probabilities in Team Persuasion Games) *Let* $\left(\mathcal{F}_{TP}, \overline{S}_0\right)$ *be the consensus version of a* ***synchronous*** *team persuasion game as in Definition 4.3.2 with balanced agents on a bipartite* $AF = (A, R)$ *with initial colouring* $\overline{S}_0$*. Let* H_{TP} *be the weighted adjacency matrix of game* $\left(\mathcal{F}_{TP}, \overline{S}_0\right)$*, and* G_{TP} *the digraph associated to* H_{TP}*.*

Each set $K \in \mathcal{K}_{\{\boldsymbol{t}\}}$ *has a value* k *that stands for the greatest common divisor (*gcd*) of the lengths of all cycles in* K*. This generates a* k*-partite graph with partitions* $\{K^1, \ldots, K^k\}$ *as in Proposition 3.3.5 . Let* μ *be a stationary distribution of* H_{TP}*, normalised such that, for each SCC* K *of* G_{TP} *we have* $\sum_{a \in K} \mu(a) = k(K)$ *Considering that* τ *stands for the duration of this game, we have that*[viii]

$$Pr\left(\overline{S}_\tau = \gamma_{for} \mid S_0\right) = \prod_{K \in \mathcal{K}_{\{\boldsymbol{t}\}}} \prod_{i=1}^{k} \left(\sum_{a \in K^i} \mu(a) \overline{S}_0(a) \right). \tag{4.5}$$

[viii] We have abused notation here: we have considered γ_{for} to be a state configuration not on the entire AF, but just on the subgraph induced by the arguments that have a path to the topic. In other words, we exclude arguments that do not even indirectly influence the acceptability of the topic.

EXAMPLE 4.4.11. Consider the synchronous game on the bipartite $AF = (V, E)$ in Figure 4.1 and S_0 as in Figure 4.4. Here, all agents $a \in A \setminus \{\mathbf{t}\}$ are 1-agents with $\mathbf{w}_1 = 1$ uniformly distributed across the set a^-. The condensation graph can be seen in Figure 4.5, so $\mathcal{K} = \{\{\mathbf{t}\}, K_1, K_2, K_3, K_4\}$, where $K_1 = \{v_{11}, v_{12}, v_{21}, v_{23}\}$, $K_2 = \{v_{13}, v_{22}\}$, $K_3 = \{v_{24}\}$ and $K_4 = \{v_{25}\}$. K_1 is the only source component. Since K_1 (indirectly) influences the acceptability of the topic, we have $\mathcal{K}_{\{\mathbf{t}\}} = \{K_1\}$. We now need to evaluate μ, a stationary distribution of the matrix H_{TP}. Then, we have

$$\mu H_{\text{TP}} = \mu \Leftrightarrow \mu \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \end{pmatrix} = \mu \Rightarrow \mu = \frac{1}{5}(1, 4, 3, 2). \tag{4.6}$$

Note that $k = 2$. We now use the initial configuration S_0 and the generalised consensus version of it, $\overline{S}_0$, according to Equations 4.2 and 4.3. We have $\overline{S}_0(v_{11}) = 0$, $\overline{S}_0(v_{12}) = 1$, $\overline{S}_0(v_{21}) = 1$, $\overline{S}_0(v_{23}) = 0$, therefore, by Theorem 4.4.10, we have

$$Pr\left(\overline{S}_\tau = \gamma_{\text{for}} \mid S_0\right) = \frac{12}{25} = 0.48. \tag{4.7}$$

Therefore, the probability of the topic being accepted is 0.48. Analogously, the probability of the topic being rejected is given by

$$Pr\left(\overline{S}_\tau = \gamma_{\text{ag}} \mid S_0\right) = \frac{2}{25} = 0.08. \tag{4.8}$$

The probability for this game not reaching a state-stable configuration is 0.44.

4.4.2.3 Probabilities in Asynchronous Games

For asynchronous games, we do not have losing configurations. That comes from the fact that now the only absorbing states are the (generalised) consensus ones. We are going to use a result from linear voting models [18, Theorem 5] that says that the mean matrix for asynchronous consensus games is given by

$$H_a = \frac{n-1}{n} I + \frac{1}{n} H_{\text{TP}} \tag{4.9}$$

Here $n = |G_{\text{TP}}| = |A|$, and I denote the $(n \times n)$ identity matrix. It is not hard to see that the mean matrix should be given by H_a: at each round, there is a probability of $\frac{n-1}{n}$ that a given agent is not chosen by the scheduler, and thus simply keeps its colour. Note that this is not the same as synchronous game in a digraph with adjacency matrix given by H_a, because in that case we could potentially have more than one node acting at the same time. That is why we used [18, Theorem 5].

Theorem 4.4.12 *Let* $\left(\mathcal{F}_{TP}, \overline{S}_0\right)$ *be the consensus version of a* ***asynchronous*** *team persuasion game as in Definition 4.3.2 with balanced agents on a bipartite* $AF = (A, R)$ *with initial colouring* $\overline{S}_0$*. Let* H_{TP} *be the weighted adjacency matrix of game* $\left(\mathcal{F}_{TP}, \overline{S}_0\right)$*, and* G_{TP} *the digraph associated to* H_{TP}*. Then,*

$$Pr\left(\overline{S}_\tau = \gamma_{for} \mid S_0\right) = \prod_{K \in \mathcal{K}_{\{t\}}} \left(\sum_{a \in K} \mu(a)\overline{S}_0(a)\right). \tag{4.10}$$

Proof. It is enough to note that although the eigenvalues differ, the eigenvectors (in particular, the stationary distribution) of H_a and H_{TP} are the same. ∎

As an illustration of the effect of asynchronicity, we are solving for the same AF.

Example 4.4.13. Consider a game as described in Example 4.4.11 with the difference that now agents act asynchronously. Let us check that μ of H_a is indeed equal to the one given in Equation 4.6.

$$\mu H_a = \mu \Leftrightarrow \mu \begin{pmatrix} \frac{3}{4} & 0 & \frac{1}{4} & 0 \\ 0 & \frac{3}{4} & \frac{1}{8} & \frac{1}{8} \\ 0 & \frac{1}{4} & \frac{3}{4} & 0 \\ \frac{1}{8} & \frac{1}{8} & 0 & \frac{3}{4} \end{pmatrix} = \mu \Rightarrow \mu = \frac{1}{10}(1, 4, 3, 2). \tag{4.11}$$

We have $\overline{S}_0(v_{11}) = 0$, $\overline{S}_0(v_{12}) = 1$, $\overline{S}_0(v_{21}) = 1$, $\overline{S}_0(v_{23}) = 0$, therefore, by Theorem 4.4.12, we have

$$Pr\left(\overline{S}_\tau = \gamma_{\text{for}} \mid S_0\right) = 0.7 \tag{4.12}$$

Therefore, the probability of the topic being accepted is 0.7. Analogously, the probability of the topic being rejected is given by

$$Pr\left(\overline{S}_\tau = \gamma_{\text{ag}} \mid S_0\right) = 0.3 \tag{4.13}$$

4.5 Bribery in Team Persuasion

In Section 4.3, we have defined the basic setup of team persuasion games, and motivated how each agent probabilistically updates the state of its arguments in each round. In Section 4.4, we provided a solution for what it means for a team to win given the acceptability of the topic, and the probability for each team to reach its goal state (i.e., Questions **D1** and **D2**). We now consider: what if in between rounds, an external agent who is not part of the game can choose to bribe one of the agents in the game to change the status of its argument (i.e. from *on* to *off* or *off* to *on*). We now motivate and answer Questions **D3** and **D4** from Section 4.1.

REMARK 4.5.1. All examples in this section assume all agents are 1-agents with $\mathbf{w}_1 = 1$ and distributed uniformly across its attackers. We further assume games are synchronous. However, generalisations for games in which there are no independence of partitions (asynchronous games or synchronous games with $\mathbf{w}_i > 0$ for some even i.), are immediate (see Remark 4.5.6).

4.5.1 Motivating Example

Consider the AF in Figure 4.6, where t is omitted but still defended by partition P_{for}. Nodes are labelled v_{11}, ..., v_{15} and v_{21}, ..., v_{26}. The entire AF forms one SCC K with $\gcd(K) = 2$ (so $P_{\text{for}} = P_1$) whose row-normalised in-matrix using this order of nodes is

$$H_{\text{TP}} = \left(\begin{array}{ccccc:cccccc} 0 & 0 & 0 & 0 & 0 & \frac{1}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \hdashline \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{3} & 0 & 0 & \frac{1}{3} & \frac{1}{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right) \tag{4.14}$$

The corresponding stationary distribution, μ, can be calculated from H_{TP}.

$$\mu = \frac{1}{506}(\underbrace{138, 72, 174, 66, 56}_{\text{for the argument}}, \underbrace{46, 72, 101, 147, 126, 14}_{\text{against the argument}}). \tag{4.15}$$

We have chosen the normalisation constant to be 506 because we would like the sum of the components of μ in each partition to be 1.[ix] In this example, 506 is half of the sum of components (= 1012). We have labelled each node in Figure 4.6 with

[ix]Here we are using the result that the sum of each partition in a given SCC is the same.

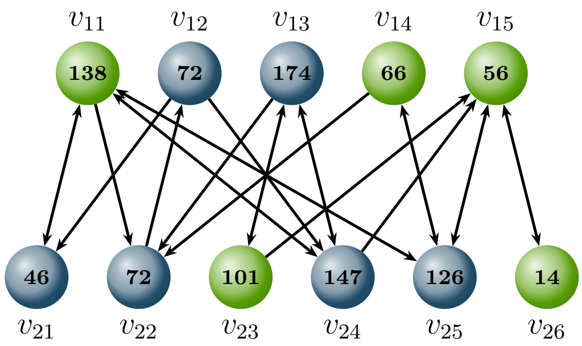


Figure 4.6: The *AF* Underlying our Example. Current Colouring and Influences Depicted in Each Argument. Influences were Multiplied by 506 for readability.

the corresponding numerical value of μ (multiplied by 506 for readability) . Note that the topic has been omitted. This is because we are interested in the probability of converging to state-stable configurations, thus which particular arguments of P_{ag} attack the topic is irrelevant. We now calculate the probability $Pr(S_\tau = \gamma_{\text{for}} \mid S_0)$ of consensus in favour of the topic being reached for this example. .

(i) In Figure 4.6, we are given $\overline{S}_0$ directly (not through an S_0). This colouring assigns *green* to v_{11}, v_{14}, v_{15}, v_{23}, and v_{26}, and assigns *blue* to all other nodes.

(ii) Figure 4.6 is already strongly connected. As the topic is attacked by some argument in P_2, we have that $\mathcal{K}_{\{t\}} = \{v_{11}, \ldots, v_{15}, v_{21}, \ldots, v_{26}\} =: \{K\}$.

(iii) Equations 4.14 and 4.15 have calculated H_{TP} and μ, respectively. Also, $gcd(K) = 2$.

(iv) Recall the notion of influence of a node (Definition 3.3.8). Here, Θ_1^g and Θ_2^g stand for the sum of influences of nodes currently coloured *green* in partitions 1 (P_{for}) and 2 (P_{ag}), respectively. Analogously, Θ_1^g and Θ_2^g stand for the sum of influences of nodes currently coloured *blue* in partitions 1 (P_{for}) and 2 (P_{ag}), respectively. We have

$$\Theta_1^g = \frac{138 + 66 + 56}{506} = \frac{260}{506} \tag{4.16}$$

and

$$\Theta_2^g = \frac{101 + 14}{506} = \frac{115}{506} \tag{4.17}$$

(v) Applying Theorem 4.4.10 and recalling that we have normalised over each partition so $\Theta_i^g + \Theta_i^b = 1$ for $i = 1, 2$, we have

$$Pr(S_\tau = \gamma_{\text{for}} \mid S_0) = \left(\frac{260}{506}\right)\left(\frac{115}{506}\right) = \frac{325}{2783} \approx 0.12 \tag{4.18}$$

Now suppose that there is an external agent G who would like the topic to win. G has the power to successfully bribe any agent in the team persuasion game to change the state of their argument, i.e. from *off* to *on* or vice versa. Equivalently, this changes the colour (*blue* or *green*) of the same node, depending on which partition it is in. We assume this bribe occurs between rounds and before all agents make their decision for the next change. Which agent should G choose to bribe? As the probability of reaching γ_{for} (and hence the topic being accepted) depends on the influences of each node, it is reasonable to conclude that G's choices on whom to bribe are between the most influential nodes in each partition that are currently *blue*. This is, either node v_{13} or v_{24}. The improvement in probabilities is given by $\frac{174 \cdot 115}{506^2} \approx 0.078$ for changing v_{13} and $\frac{147 \cdot 260}{506^2} \approx 0.149$ for changing v_{24}. Thus, counter-intuitively, although the influence of v_{13} is greater than v_{24}'s, G will have a greater improvement in their utility by changing the state of agent v_{24}. The example above motivates the following definition of the payoff of G.

Definition 4.5.2 (Single Player Utility) *We define G's **utility** of bribing an agent $a \in A$, $u_G(a)$, by the change in probability that the topic definitively wins after the agent a is successfully bribed. We define $u_G(i)$ as the change in probability of G winning given that a currently* blue *node with highest influence (it might not be unique) in partition P_i has been chosen, i.e., $u_G(i) = \max_{a \in P_i} \{u_G(a)\}$. Let B be an external agent that seeks to bribe a* green *agent to increase the probability of the topic being rejected. We define $u_B(a)$ analogously and $u_B(i)$ as the change in probability of B winning given that (one of) the most influential green nodes in partition P_i have been chosen.*

Before seeking to answer Questions **D3** and **D4** relating to the two briber problem, we first define the following notation.

n-AF: we say an AF is an n-AF iff (1) A has only one SCC; (2) AF is bipartite with partitions P_{for} and P_{ag}; and (3) if the greatest common divisor of the length of all cycles in A is n. In particular, an n-AF is also a n-partite AF. For example, in the AF depicted in Figure 4.6, the greatest common divisor is 2, thus it is a 2-AF.

$P_1, \ldots, P_n$**:** are the partitions of an n-AF such that $P_i(K) \subset P_{\text{for}}$ iff i is odd (and $P_i(K) \subset P_{\text{ag}}$ iff i is even). Note that an n-AF is both bipartite (with partitions P_{for} and P_{ag}) and n-partite (with partitions $P_1, \ldots, P_n$).

g_i **(*resp.* b_i):** is the highest influence among agents currently coloured *green* (resp. *blue*) in partition P_i, i.e,

$$g_i = \max_{a \in P_i \, and \, \overline{S}(a)=1} \{\mu(a)\} \tag{4.19}$$

$$b_i = \max_{a \in P_i \, and \, \overline{S}(a)=0} \{\mu(a)\} \tag{4.20}$$

$\widehat{\Theta^g_I}$**:** is the product of Θ_i, $0 \leq i \leq n$ such that $i \notin I$, where $I \subset \{1, \ldots, n\}$. e.g., $\widehat{\Theta^g} = \prod_{k=1}^{n} \Theta_k$, or $\widehat{\Theta^g_i} = \prod_{k=1, k\neq i}^{n} \Theta_k$. We define $\widehat{\Theta^b_I}, \widehat{b_I}$, and $\widehat{g_I}$ analogously. We omit the curly brackets from the set I (e.g., $\Theta_{i,j}$) for readability.

Example 4.5.3. In the example in Section 4.5.1, we have a 2-AF, partitions P_1 and P_2, $g_1 = \frac{138}{506}$, $g_2 = \frac{101}{506}$, $b_1 = \frac{174}{506}$, $b_2 = \frac{147}{506}$, $\widehat{\Theta^g} = \frac{325}{2783}$, and finally $\widehat{\Theta^b} = \frac{2091}{5566}$.

4.5.2 The Case of a Single Briber

We formalise and answer Question **D3** by presenting the following Lemma.

Lemma 4.5.4 (Bribery - Single Player) *Consider a team persuasion game in an n-AF. Let G be an external agent willing to bribe one currently* blue *agent. Under these conditions, in order to bribe an agent, G is willing to pay (subject to their risk profile) at most*

$$\max_{0 \leq i \leq n} \{u_G(i)\} = \max_{0 \leq i \leq n} \{b_i \widehat{\Theta^g_i}\} \tag{4.21}$$

In other words, G will choose one from most influential nodes in each partition to bribe. Analogously for agent B, we have

$$\max_{0 \leq i \leq n} \{u_B(i)\} = \max_{0 \leq i \leq n} \{g_i \widehat{\Theta^b_i}\}. \tag{4.22}$$

Proof. We just have to show that $u_G(i) = b_i \widehat{\Theta^g_i}$. Indeed, $u_G(i) = (b_i + \Theta^g_i)\widehat{\Theta^g_i} - \widehat{\Theta^g} = b_i \widehat{\Theta^g_i}$. ■

	P_1	P_2
P_1	1.6% , −5.5%	−9.3% , −23.7%
P_2	0.9% , −1.0%	4.7% , −4.4%

Table 4.1: A 2-Player Game. G plays in rows and B in columns. Cells denote payoff of G (left) and B (right).

EXAMPLE 4.5.5. Consider the example from Section 4.5.1 from the perspective of agent B, who is seeking to bribe a *green* agent. B's options are either node v_{11} or v_{23} given that they are the most influential nodes of their partitions. Applying Lemma 4.5.4, B is willing to pay at most $\max_{0 \leq i \leq 2}\{g_i \widehat{\Theta_i^b}\} = \max\{\frac{51}{242}, \frac{24846}{506^2}\} = \frac{51}{242}$.

REMARK 4.5.6. The solution for the briber in asynchronous games or synchronous games with $\mathbf{w}_i > 0$ for some even $i > 1$, is to simply choose the agent with highest influence currently not in the desired state.

4.5.3 The Case of Two Bribers

What if there are two competing external agents bribing nodes simultaneously? We answer this question by considering two external agents G and B. As before, G is to bribe a *blue* agent in order to increase the probability of the topic being accepted. Also, B is to bribe a *green* agent in order to increase the probability of the topic being rejected. Note that their options do not overlap, because a node is never *blue* and *green* at the same time. Let us look back to our motivating example from Section 4.5.1, in which G had the choice between bribing v_{13} or v_{24}. Since v_{11} and v_{23} are the most influential nodes currently *green* in partitions P_{for} and P_{ag} then B should choose to bribe either one of these agents, because the effect of these agents changing their colour gives the largest change in the probability of B obtaining a desirable outcome. We assume both bribes from G and B occur simultaneously. Which of the agents should each briber choose? Table 4.1 depicts the payoff of each scenario.

We can see that strategy (P_1, P_1) is a pure strategy Nash equilibrum (PSNE) of this game, and also the only one, since no agent can benefit from changing their strategy while the other agents' strategies remain the same. We then expect G to have an increase of 1.6% on their probability of winning, whereas B will have their probability decreased by 5.5%. Why would B play the game in the first place if they lose? B's decision of whether to play the game is not represented as an action in

	P_1		P_k		P_n
P_1	$(b_1 - g_1)\widehat{\Theta_1}$	$\dots$	$[b_1(\Theta_k - g_k) - g_k\Theta_1]\widehat{\Theta_{1,k}}$	$\dots$	$[b_1(\Theta_n - g_n) - g_n\Theta_1]\widehat{\Theta_{1,n}}$
	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
P_k	$[b_k(\Theta_1 - g_1) - g_1\Theta_k]\widehat{\Theta_{k,1}}$	$\dots$	$(b_k - g_k)\widehat{\Theta_k}$	$\dots$	$[b_k(\Theta_n - g_n) - g_n\Theta_k]\widehat{\Theta_{k,n}}$
	$\vdots$	$\ddots$	$\vdots$	$\ddots$	$\vdots$
P_n	$[b_n(\Theta_1 - g_1) - g_1\Theta_n]\widehat{\Theta_{n,1}}$	$\dots$	$[b_n(\Theta_k - g_k) - g_k\Theta_n]\widehat{\Theta_{n,k}}$	$\dots$	$(b_n - g_n)\widehat{\Theta_n}$

Table 4.2: 2-player Game on a Bipartite Graph. G Plays in Rows and B in Columns.

the payoff matrix, but rather is assumed to have been made by B prior to the game. One way of endogenising this information is to introduce a prior decision whether each agent will play the game, and their expected utility either way. A simpler way, however, is to say that B is willing to be paid (instead of pay) at least 5.5% utility in order to play this game. The following definition formalises how we calculate the payoff on the 2-player game in an *n-AF* and Table 4.2 can be used for quick reference.

Definition 4.5.7 (2-Player Utility) *Consider a team persuasion game in an n-AF with current configuration s_j. Let G and B be external agents bribing a blue or green argument, respectively. We define $u_G(i,j)$ (resp. $u_B(i,j)$) as the utility function for player G (resp. R) given that G has chosen partition i and B partition j to bribe, s.t. $0 \leq i,j \leq n$. This function is given by the change in probability of the respective team winning, i.e,*

$$u_G(i,j) = \begin{cases} (b_i - g_i)\widehat{\Theta_i^g} & \text{if } i = j, \\ \left[b_i(\Theta_j^g - g_j) - g_j\Theta_i^g\right]\widehat{\Theta_{i,j}^g} & \text{if } i \neq j. \end{cases} \tag{4.23}$$

and

$$u_B(i,j) = \begin{cases} (g_i - b_i)\widehat{\Theta_i^b} & \text{if } i = j \\ \left[g_j(\Theta_i^b - b_i) - b_i\Theta_j^b\right]\widehat{\Theta_{i,j}^b} & \text{if } i \neq j \end{cases} \tag{4.24}$$

We are now ready to explore the following question: is it a coincidence the example showed in Table 4.1 has a PSNE? Recall that every n-player game where each player can take finitely many actions has a mixed strategy Nash equilibrium (MSNE) [55, 56]. We now prove it was not a coincidence: our two-person bribery game always has a PSNE.

Lemma 4.5.8 *A 2-player game in a 2-AF always admits at least one PSNE.*

	P_1	P_2
P_1	$(b_1 - g_1)\Theta_2^g\,,\,(g_1 - b_1)\Theta_2^b$	$b_1(\Theta_2^g - g_2) - g_2\Theta_1^g\,,\,g_2(\Theta_1^b - b_1) - b_1\Theta_2^b$
P_2	$b_2(\Theta_1^g - g_1) - \Theta_2^g g_1\,,\,g_1(\Theta_2^b - b_2) - b_2\Theta_1^b$	$(b_2 - g_2)\Theta_1^g\,,\,(g_2 - b_2)\Theta_1^b$

Table 4.3: 2-player game on a 2-AF. G plays in rows and B in columns.

Proof. The payoff matrix for a general 2-player game in a 2-AF is given by Table 4.3. In order for the game to not have a PSNE, we need, **wlog**, that

$$(b_1 - g_1)\Theta_2^g \leq b_2(\Theta_1^g - g_1) - \Theta_2^g g_1, \quad \textit{and} \tag{4.25}$$

$$(b_2 - g_2)\Theta_1^g \leq b_1(\Theta_2^g - g_2) - \Theta_1^g g_2. \tag{4.26}$$

However, this condition leads to $b_1 g_2 \leq -b_2 g_1$, a contradiction since influences are all positive. In order to avoid the existence of a PSNE we would need a player that deviates from both diagonal cells, and that can never be the case as shown above.■

We now explore the scenario in which there are two PSNE in a 2-AF. We consider that, in case one of the equilibria is as good as the other for both players, this will be the one to determine the expected utility. However, if not, we then consider the expected payoff of the MSNE in which both partitions are chosen with positive probability by both G and B. The next proposition evaluates this outcome.

Proposition 4.5.9 (Expected utility from mixed strategy in 2-*AF*s) *Let AF be a 2-AF in which the 2-person game has two PSNE. Then, the expected utility for G is given by*

$$\mathbb{E}(u_G) = -\Theta_1^g\Theta_2^g + \frac{1}{g_1 b_2 + g_2 b_1}\left[-b_1 b_2 g_1 g_2 + \sum_{i=1}^{2} \Theta_i^g \widehat{b_i}\widehat{g_i}(\Theta_i^g + b_i - g_i)\right] \tag{4.27}$$

For B, we get $\mathbb{E}(u_B)$ by swapping b_k for g_k and Θ_k^g for Θ_k^b in in the above formula.

The proof of the above proposition is given by direct calculation of the MSNE. We now consider the case in which we have an n-AF. Is it the case that there is also always a PSNE? The following theorem proves that indeed, a PSNE is always present and, if unique, describes the expected utility for each player. We leave the evaluation of the MSNE in the non-unique case for future work.

Theorem 4.5.10 *A 2-player game in an n-AF always admits at least one PSNE.*

Proof. We prove this by contradiction by applying results in [53, Corollary 2.2 and Theorem 2.8]. All we have to do is show that there is no set of four strategy profiles of the form $(l, u), (r, u), (l, d), (r, d)$ (i.e., forming a rectangle in Table 4.3, where l denotes the left column, u the upper row, and so on) such that the four following equations cannot simultaneously hold:

$$u_B(r, u) > u_B(l, u) \tag{4.28}$$

$$u_G(r, d) > u_G(r, u) \tag{4.29}$$

$$u_B(l, d) > u_B(r, d) \tag{4.30}$$

$$u_G(l, u) > u_G(l, d) \tag{4.31}$$

Assume by contradiction that we do have such a rectangle. We split the proof in three cases and show that each of them lead to contradiction. Each case considers a different number of strategies laying on the diagonal of 4.2, which can be:

(a) None; or

(b) Exactly one; or

(c) Exactly two.

For Case **(a)**, on one hand, from Equation 4.29 we get $g_u\Theta_d > g_d\Theta_u$. On the other hand, from 4.31 we get $g_u\Theta_d < g_d\Theta_u$, thus we have a contradiction (that could also have been derived from the other two equations). For Case **(b)**, we can consider **wlog** that $r = d$. From Equation 4.28 we get $b_l\Theta_d > b_d\Theta_l$. Also, from Equation 4.30 we get $b_d\Theta_l - b_l\Theta_d - g_db_l > 0$. Combining both, we get a contradiction by observing that $0 < b_d\Theta_l - b_l\Theta_d - g_db_l < b_l\Theta_d - b_l\Theta_d - g_db_l = -g_db_l$. For **(c)** we use Equations 4.28 and 4.30 and proceed analogously to proof of Lemma 4.5.8. ∎

4.6 Related Work

In this chapter we have presented and analysed an argumentation model for a very common form of public debate. Our work has made two novel contributions. The first contribution is the formalisation using argumentation frameworks of public policy debates where multiple parties with only local information propose arguments to support (or attack) claims of interest to a wider audience, seeking to persuade that audience of a claim (or not, as the case may be). The second contribution is the use of Flag Coordination Games, specifically its analysis of the dynamics of graph colouring, to understand the properties of this formal framework. Analogues of graph colouring have been used in argumentation, for example, in labelling semantics to determine acceptability of arguments [13]. However, to the best of our knowledge, interpreting such colourings as the argument having been asserted or not, and the dynamics of how such a colouring changes, have not previously been used in argumentation theory.

The general problem of two parties with contradictory viewpoints, each seeking to persuade an impartial third party of their viewpoint, has been investigated in economics, e.g. using game theory [70, 71] or mechanism design [31, 32]. Applying argumentation theory to study multi-agent persuasion with two teams, in which one is arguing for the acceptability of a topic and the other against, has been investigated in the work by Bonzon and Maudet [10]. They focus specifically on the problem with respect to the kinds of dialogue that occur on social websites, specifying that agents "vote" on the attack relations between arguments. One of the main differences between their work and ours is that agents in their formulation do not have any motivation to act in a way that might be detrimental to their team's goal, whereas agents in our work may also be motivated by their own individual goals. In the context of bipartite graphs, the problem of determining the acceptability of a specific argument, for both credulous and sceptical semantics, has been shown to be decidable in polynomial time by Dunne in [26].

Dignum and Vreeswijk developed a testbed that allows an unrestricted number of agents to take part in an inquiry dialogue [23]. The focus of their work is on the practicalities of conducting a multi-party dialogue, concerned with issues like turn-taking, rather than in the strategising of agents participating in such a dialogue. Bodanza *et al.* [9] survey work on how multiple argumentation frameworks may be aggregated into a single framework. While this direction of work considers how frameworks from multiple agents might be merged, it removes the strategic aspect of persuasion which we are interested in here. Building on the idea of the strength of an argument, which has been in discussion since at least 1995 through work by Krause *et al.* [45], Dunne *et al.* present a framework of weighted argument systems in [28].

There is an established literature that applies ideas from game theory to argumentation. For example, in using zero-sum two-player games to assign strengths to arguments satisfying intuitive properties [49], or studying the strategy-proofness of the grounded extension in the context of mechanism design [62]. Both these works focus on the actions of the agents engaging in the dialogue rather than the actions of external bribers that can influence agents in the dialogue. Unlike the setup of [49, 62], the assumption in team persuasion games that each agent only proffers one argument may be seen as too restrictive. Future work can investigate how agents in team persuasion games can proffer multiple arguments within, or perhaps across, a partition. Dialogues have been studied in a game-theoretic perspective in order to identify Nash Equilibria [40], however, unlike us, they consider the game from the perspective of the interlocutors, not the bribers.

4.7 Summary of Results

We have shown how to determine the probability of each team winning in a team persuasion game (Question **D2**), both when agents act synchronously (Theorem 4.4.10), and asynchronously (Theorem 4.4.12). Although we have depicted all consensus states (Question **D1**), we have shown that not all synchronous games become state-stable, having no definite winner.

We have conducted a game-theoretic analysis of how external agents can and should bribe the agents of the game. We believe this is the first work that considers the issues of bribery in dialogical argumentation. We have considered how external agents in team persuasion games can interact with the interlocutors to influence the outcome of the game in their favour. Specifically, we have derived expected utilities for a briber in both the single-briber (Question **D3**) and two-briber (Question **D4**) scenarios. In future work, the results for team persuasion games can be applied to other types of argument dialogue games, such as negotiations [63]. While team persuasion games are similar to real-world political debates where bribery is common, there are other forms of dialogue where it might also occur.

Chapter 5

Biased Consensus Games

Problem 5 (Biased Game on a Cycle). Consider the 17-cycle in Figure 5.1. As this is a non-bipartite graph, the theorem by Hassin and Peleg (Theorem 2.3.1) gives us the probabilities involved in this game: $Pr(\bullet) = \frac{8}{17}$ and $Pr(\bullet) = \frac{9}{17}$. Now consider that there is a bias towards opinion blue. In particular, consider that the decision algorithm of a node that currently lies between a blue node and a red node consists of copying blue with a probability of $\frac{2}{3}$ and red with a probability of $\frac{1}{3}$. We can say that blue has an advantage under these circumstances. Is blue now more likely to win than red, given a particular starting configuration, such as the one in Figure 5.1? What are the probabilities involved in this case?

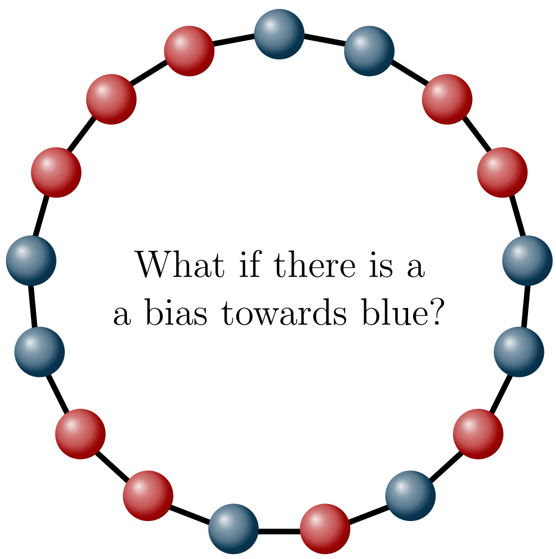


Figure 5.1: Biased Consensus Game on a 17-Cycle.

5.1 Introduction

We present Problem 5 as an abstraction of a wide range of possible concrete applications of what we will define as Biased Consensus Games.

In the context of voting algorithms, for example, consider a sequence of rounds of voting that lead to a consensus in a given opinion x. Then assume this same population is going to enter another voting game, with the same set of opinions to be chosen from as before. It is natural to consider that there will be a bias towards a given opinion in the light of their previous game. For instance, opinion x might be preferred by voters, depending on their behaviour, as it has recently won.

In the context of population genetics models, in another application domain, we consider that a new individual carrying a (x) mutation is introduced into a given population. We can model a process in which an individual will be replaced by an offspring of one of its neighbours. This neighbour is chosen taking in account its mutation's fitness.

As a refinement of Questions **A7** and **A8**, in this chapter, we will explore the following questions.

E1 Is the probability of a given colour winning a biased game a linear function with respect to the number of nodes (or edges) of that colour in the initial configuration?

E2 Is the initial relative position of nodes (of the same degree) irrelevant regarding the probabilities of winning for each colour?

E3 Is there a martingale (with respect to the random variable $\{S_t\}_{t\geq 0}$ that describes the game) that, together with Doob's sampling theorem (Theorem 2.4.15), gives us the probabilities of consensus given the initial configuration?

E4 How do we further develop and prove the idea in Section 3.2.1 that connects Flag Coordination Games and processes involving random walks?

E5 Given a game $(\mathcal{F}, S_0)$, and a state $s \in \mathcal{S}$, is $Pr(S_t = s \mid S_0) > 0$ for some $t \geq 0$? This is the reachability problem presented in Question **A8**.

E6 Given a bias towards colour x, what is the minimum number of nodes of that colour we need in order for x to be more likely to win than not? Conversely, given a number of nodes coloured x in a given graph, what is the minimum bias towards x for which x is still more likely to win than not? In other words, we are looking at what sort of trade-off there are between bias and bias towards colour x and the amount of nodes coloured x in a given graph.

E7 How can we formally define a sequence of multiple iterations of similar games in which biases towards each colour might change from one game to the next depending on the consensus achieved in previous iterations?

In this chapter and dissertation, we are only looking into these questions in the context of games played on cycles. To exploring other structures is left for future work.

Note that the answers to Questions **E1**, **E2**, and **E3** are affirmative in the context of *unbiased* games on odd cycles, or more generally in the context of unbiased games on regular undirected non-bipartite graphs.[i] The regularity and the lack of partition asymmetry imply positive answer for Questions **E1** and **E2** (recall Theorem 2.3.1). Regarding Question **E3** we do not even need the use of regularity, since Theorem 2.3.1 guarantees that a martingale exists for any unbiased game on undirected non-bipartite graph.

Our main objective in this chapter is to study whether these properties (related to Questions **E1**, **E2**, and **E3**) also hold for the biased version of the consensus game on cycles when only two colours are involved. Initially, we are restricting ourselves to cycles of odd length, so there are no losing configurations involved. However, earlier results (Theorem 3.2.26) will allow us to immediately generalise the results for even cycles. This analysis will only be possible with results related to Question **E4**, which will, in turn, play a key role when exploring Question **E5**.

This chapter is structured as follows: Section 5.2 briefly introduces the related work pertinent to this chapter. In Section 5.3 we provide a formal definition of biased consensus games to then give the results (Questions **E1**, **E2**, and **E3**) for such games on cycles (Section 5.3.1), using results related to Question **E4** (Section 5.3.1.1). In Section 5.4.1, we answer Question **E5** for cycles, whereas in Sections 5.4.2 and 5.4.3, we motivate and formally define families of problems related to Questions **E6** and **E7**, respectively.

[i]Recall that an undirected graph is r-regular if for all $v \in V$, $\deg v = r$.

5.2 Related Work

We focus our attention first on a similar model initially proposed by Patrick Moran in 1958 [54] in the context of population genetics models. Lieberman *et al.* (see [48]) present a generalisation of Moran Processes with the use of weighted directed graphs to represent the population individuals (nodes) and the probabilities that each neighbour is chosen to be replaced (edges). More precisely, the process can be described as follows.

(i) A population of fixed size is represented by a weighed directed graph $G = (V, E)$.

(ii) At each step an individual $v \in V$, is chosen proportional to its fitness. Then v reproduces placing its offspring as a replacement of a given neighbouring node $w \in \mathcal{N}(v)$ with probability according to the weighted edge $e = (v, w) \in E$.

The case where G is a complete graph with equally weighted edges represents the original Moran Process. Note that this process differs from Biased Consensus games in the following ways:

(i) The generalised Moran Process described above is asynchronous, in which one node acts in each time step, whereas Biased Consensus Games consider that all agents act at the same time.

(ii) Moran Processes, according to the way they have been described, are not Flag Coordination Games since the replaced node w is not independently choosing its state, but instead being replaced as a result of the decision of its neighbour v.

Consider a Moran Process in complete graph size n in which all edges are equally weighted and suppose all resident individuals are identical and one new mutant is introduced (see [48]). The mutant has relative fitness r, whereas residents have fitness 1. Then, the fixation probability of this new mutant is

$$\rho = \frac{1 - \frac{1}{r}}{1 - \frac{1}{r^n}} \tag{5.1}$$

Lieberman *et al.* provide solutions for other graph structures, such as the star graph (ρ_2), and the super-star (ρ_k), which are given by

$$\rho_2 = \frac{1 - 1/r^2}{1 - 1/r^{2n}} \qquad \text{and} \qquad \rho_k = \frac{1 - \frac{1}{r^2}}{1 - \frac{1}{r^{kn}}}. \tag{5.2}$$

5.3 Formal Definitions and Results

We have seen in Section 3.2.4 that in generalised consensus games (see Definition 3.2.1) we can assign different weights to different nodes in the graph, and that the impact of this change on evaluating probabilities of convergence is minimal: the proof of Theorem 3.3.16 is essentially the same regardless of nodes having a bias towards a given neighbour, as long as this is a constant bias (recall discussion in Section 3.2.4).

What we explore in this Chapter, however, is something else. We analyse the impact of bias on colours (or opinions) on the probability of a given opinion to win a (generalised) consensus game.

We now formally define a biased consensus game in a general graph $G = (V, E)$. Later, we will focus our analysis on particular graph classes.

Definition 5.3.1 (Biased Generalised Consensus Game) *We define the rules of a biased consensus game $\mathcal{F}_\Delta = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$ as in Definition 3.2.1, with the difference that we introduce bias into agents' decision algorithms $\mathcal{A}$. Let X be the set of colours in this game, with colour biases $\delta_1, \delta_2, \ldots, \delta_{|X|} \in \mathbb{R}$.*[ii] *Also, for each node $v \in V$, let $|\mathcal{N}_x(v)|$ be the number of neighbours of v that are coloured x in the current configuration S_t. Finally, note that nodes act synchronously. Then, for a given v, we have*

$$Pr(S_{t+1}(v) = x \mid S_t) = \frac{\delta_x \, |\mathcal{N}_x(v)|}{\sum_{i=1}^{|X|} \delta_i \, |\mathcal{N}_i(v)|}. \tag{5.3}$$

Remark 5.3.2. We can also think of biased consensus games as each node having an urn in which they place δ_x balls of colour x for each neighbour currently coloured x, and then drawing one ball uniformly at random from the urn. The only remark to this analogy is that we allow, for generality, the biases to be real numbers.

Remark 5.3.3. Note that we are somewhat abusing notation in Definition 5.3.1 if we want it to allow include generalised biased consensus games. In order to consider generalised games, it is enough to replace the biases towards colours to biases towards winning colourings $\gamma \in \Gamma$. As before (see Definition 3.2.1), we require that the goal states can be uniquely identified from the colour any given node.

[ii] To guarantee uniqueness, we can also define that $\sum_{i=1}^{|X|} \delta_i = 1$. or, alternatively, that $\delta_1 = 1$. Instead, however, we will give up uniqueness to improve readability of results later in this chapter.

Given an initial configuration of the game, we are interested in the probability that the consensus is achieved for each colour x. Note that, for biased consensus games on general graphs, such probilities are likely to be hard to find. That is because of its similarities with Moran-like processes on a general graph and the fact that such problems are PSPACE-Hard [37].[iii] In this dissertation, we are not making use of this result in any way but to indicate the hardness of finding analytic solutions for the probabilities of consensus in biased consensus games.

That is the main reason why we are, from this point onward, going to focus only on graphs that are cycles. Although there might exist other graph structures for which an analytic solution is likely to be found (such as paths), this is beyond the scope of this dissertation and will be left for future work.

5.3.1 Biased Games on Cycles with Two Colours

In this section, we will explore biased games on cycle graphs C_n. The following formal definition will help us easily refer to this instance of Flag Coordination Games throughout the rest of this chapter.

Definition 5.3.4 (Biased Two-colour Consensus Game on $\boldsymbol{C_n}$) *We say that* $\mathring{\mathcal{F}}_\Delta = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ *is the set of rules of a biased two-colour consensus game on a cycle if it satisfies Definition 5.3.1, and we have* $G = C_n$*,* $X = \{●, ●\}$ *and biases* $\delta_{blue} = b$ *and* $\delta_{red} = r$*.*

We first answer Question **E1** by means of a counterexample. This is, linearity must not be a general property for biased consensus games on cycles, as in Example 5.3.5 shows that probabilities are not linear with respect to the number of nodes of a given colour on C_3.

Example 5.3.5 (Biased Game on C_3). Let $(\mathring{\mathcal{F}}_\Delta, S_0)$ be a biased two-colour consensus game on a cycle as in Definition 5.3.4 with $G = C_3$. By symmetry there are only four different configurations $\beta_0, \ldots, \beta_3$, where β_i represents a configuration in which there are i blue nodes, as depicted in Figure 5.2. We define B_i as the probability of colour *blue* winning the consensus game, given that the current configuration has i *blue* nodes. For example, $B_0 = 0$ and $B_3 = 1$. Further, we combine the two relations:

[iii] PSPACE is the set of decision problems that, using a polynomial amount of space, can be solved by a (deterministic or not) Turing Machine. PSPACE-Hard refers to the set of problems that can be reduced to from all problems in PSPACE.

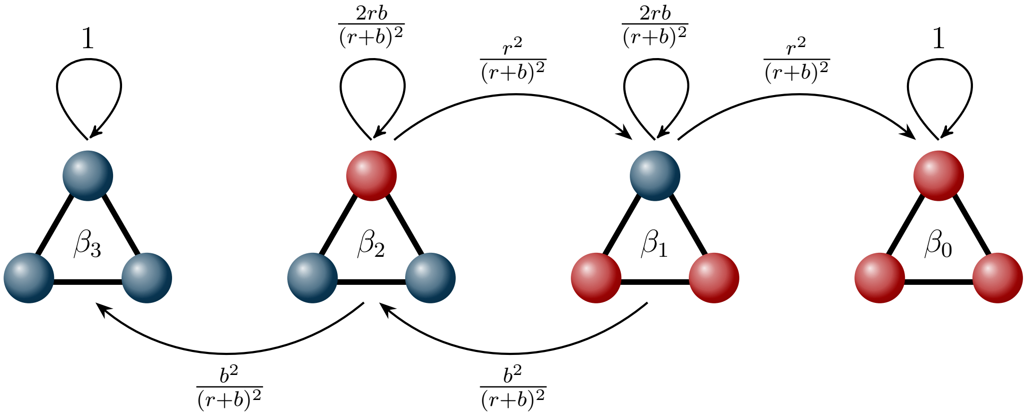


Figure 5.2: Possible States and Their Transition Probabilities of a Biased Consensus Game on C_3.

$$
\begin{aligned}
B_1 &= \frac{r^2}{(r+b)^2} B_0 + \frac{2rb}{(r+b)^2} B_1 + \frac{b^2}{(r+b)^2} B_2 && (5.4)\\
B_2 &= \frac{r^2}{(r+b)^2} B_1 + \frac{2rb}{(r+b)^2} B_2 + \frac{b^2}{(r+b)^2} B_3. && (5.5)
\end{aligned}
$$

Using $B_0 = 0$ and $B_3 = 1$, we get

$$B_1 = \frac{b^4}{b^4 + b^2 r^2 + r^4} \qquad B_2 = \frac{b^4 + b^2 r^2}{b^4 + b^2 r^2 + r^4}. \tag{5.6}$$

Also, the average probability of blue winning, considering all 8 possible initial states with equal probability, is given by

$$\frac{49r^4 + 25r^2 b^2 + b^4}{8(r^4 + b^4 + r^2 b^2)}. \tag{5.7}$$

Based on Equation 5.6, we can conclude that the answer to Question **E1** is **negative** for biased consensus games on cycles.

Note that the relation $B_i + A_{n-i} = 1$ always holds, where A_i denotes the probability of colour *red* winning the consensus game. However, the relation $B_i + A_{n-i} = 1$ (which is equivalent to $B_i = A_{n-i}$) does not hold for a general set of biases Δ on cycles. This fact comes from the lack of linearity dealt in Question **E1**. Indeed, only for the situation in which $b = r$ is that we have $B_i = R_{n-i}$ on cycles. In particular for C_3 we have the expected results in line with Theorem 2.3.1: $B_1 = \frac{1}{3}$ and $B_2 = \frac{2}{3}$.

As an illustration of the impact of bias, applying the results from the example above (Example 5.3.5) for the particular case in which $b = 2r$, we have $B_1 = \frac{16}{21}$ and $B_2 = \frac{20}{21}$. We are going to use this particular results to show in Remark 5.3.6 that Moran processes and biased consensus games are different.

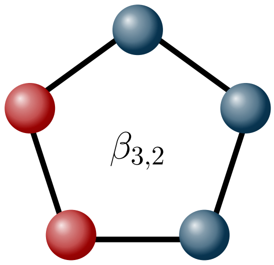


Figure 5.3: Configuration $\beta_{3,2}$ of C_5.

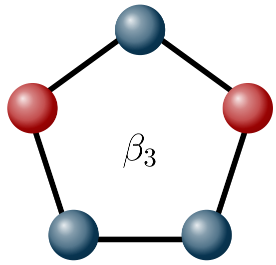


Figure 5.4: Configuration β_3 of C_5.

Remark 5.3.6. Note that $C_3 = K_3$, so Equation 5.3 holds and we can then compare the results between the biased game (with $r = 1$) and a Moran process (with mutation fitness b) in the same graph, with same initial configuration β_2. According to Equation 5.3, we have

$$\rho = \frac{4}{7} \neq \frac{16}{21} = B_1 \tag{5.8}$$

Thus, although similar, both processes are not equivalent.

The example with C_3 does not allow us to establish whether the initial relative positions of the same number of nodes of a given colour affect the result or not (Question **E2**). However, we do not have to go too far to get non-isomorphic configurations (considering nodes' colours) with the same number of nodes of a given colour. In order to help us to investigate Question **E2**, we calculate the probability of the two initial configurations for the game in C_5 as in Figures 5.3 and 5.4. As before, $B_3, 2$ and B_3 represent the probabilities that games $\beta_3, 2$ and β_3, respectively, converge to consensus in *blue*.

Direct calculation gives us the same result for both initial cases:

$$B_{3,2} = B_3 = \frac{b^8 + b^6r^2 + b^4r^4}{b^8 + b^6r^2 + b^4r^4 + b^2r^6 + r^8}, \tag{5.9}$$

which might suggest that the answer to **E2** is true for biased games on odd cycles. We can also conjecture that there might exist a martingale for the general problem in the cycle which resembles polynomials of the form $\sum_{j=1}^{i} b^{2(n-1-j)}r^{2(j-1)}$.[iv] We investigate the matter further by translating this problem into one involving random walks, analogous to the problem described in Chapter 3. The only difference between Definition 3.2.4 and the one generated from biased games is that, in biased games random-walking particles have different probabilities for moving clockwise or counter-clockwise. It is thus enough to provide a proof for the biased version of

[iv]Note that this is a geometric sum with ratio $\left(\frac{r}{b}\right)^2$.

these consensus games on the cycle, since a solution for unbiased ones will follow immediately from a solution for the more general biased version. We develop this relationship in the next section.

5.3.1.1 More on Annihilating Random Walks and Flag Coordination Games

We initially define the position of the random-walking particles as coinciding with the positions in which nodes are randomising. We will prove that, given an appropriate analogy between the two random choices, the position of random-walking particles will still match with randomising nodes on the following round, and therefore for the entire process.

We first define the process with random-walking particles independently to subsequently connect both.

Definition 5.3.7 (Annihilating Biased Random Walks on a Cycle) *Let C_n be a cycle with n nodes, n odd, and assume there are initially $2m < n$, with $m \in \mathbb{N}$, particles performing a biased random walk on this cycle (i.e., each particle has a constant probability p of moving clockwise and $q = 1 - p$ of moving counter-clockwise). All the particles move synchronously. If, in the end of any round, two particles meet at the same node, both disappear. In order to facilitate the description of this process, we will also colour each particle according to set Y. Let R be the random process $\{R_t \mid t \in T\}$, indexed by discrete time-set T, which describes this game. Formally, $R_t : V \to ([0,1] \times Y) \cup \{-1\}$ such that*

$$R_t(v) = \begin{cases} -1 & \text{if there are no particles in } v \text{ in round } t, \\ (p, y) & \text{otherwise.} \end{cases} \tag{5.10}$$

Here p denotes the probability that this particle moves clockwise *at each round and $y \in Y$ is this particle's colour. Because there are an even number of particles and n is odd, the process will eventually end, i.e., all particles will disappear.*

In the case that n is even, we define this process as above, considering the restriction that each partition of this bipartite graph hosts an even number of particles in the initial round (and therefore also in subsequent rounds).

Remark 5.3.8. Note that games described in Definition 5.3.7 are not Flag Coordination Games, because nodes do not decide on their state in the following round. Instead, we can see this as a push-model process (e.g. as seen in [18, 48]).

We now recall Definition 3.2.4 from Chapter 3. In summary, we placed a particle on randomising nodes. Now, however, particles are performing a biased random walk, so we update the definition as follows.

Definition 5.3.9 (Correspondence Between Games 5.3.4 and 5.3.7) *Let $(\mathring{\mathcal{F}}_\Delta, S_0)$ be a biased two-colour consensus game on C_n, $n \in \mathbb{N}$. We define the function f that takes a configuration S as input and returns a configuration of random walks R, as in Definition 5.3.7 such that*

$$(f(S))(v_i) = R(v_i) = \begin{cases} -1 & \text{if } S(v_{i-1}) = S(v_{i+1}), \\ \left(\frac{b}{b+r}, red\right) & \text{if } S(v_{i-1}) \neq S(v_{i+1}) = red, \\ \left(\frac{r}{b+r}, blue\right) & \text{if } S(v_{i-1}) \neq S(v_{i+1}) = blue. \end{cases} \tag{5.11}$$

where v_{i+1} corresponds to the neighbour of v_i clockwise, and v_{i-1} corresponds to the counter-clockwise neighbour of v_i. Note that both fractions lie in $[0, 1]$, and that the number of randomising nodes in S is always even, therefore the transformation does give us a state of a process, as in Definition 5.3.7.

Example 5.3.10. Figure 5.5 illustrates two main aspects of what has been described so far. Firstly, it provides the full Markov chain and the transition probabilities for a biased game on C_5. This example differs from the one involving the graph C_3 (shown in Figure 5.2) as there are, up to symmetry, more than one configuration in which two nodes are randomising. Note that symmetric states have been identified in this illustration, and transition probabilities have been combined.

Secondly, Figure 5.5 exemplifies the position of random-walking particles on C_5 after applying the function f. Red particles are represented by p_1 and p_3, whereas p_2 and p_4 represent blue particles. Note that the annihilation of pairs of particles that occur when going from the top layer to the middle layer is only one of many possible alternatives, as well as the reorientation of particles from $\widehat{\beta}_3$ and $\widehat{\beta}_2$. However, note that the probabilities indicated correspond to the state transitions in the consensus game rather than the one in the random walks process.[v]

Proposition 5.3.11 *The function $f \restriction_{\mathcal{S}\setminus\Gamma}$ is bijective when n is odd. Here $\Gamma = \{\gamma_{blue}, \gamma_{red}\}$ is the set of goal states, where γ_{blue} and γ_{red} represent consensus in blue and red, respectively.*

[v]There might be connections between this and John Baez's work on Categorification [3]

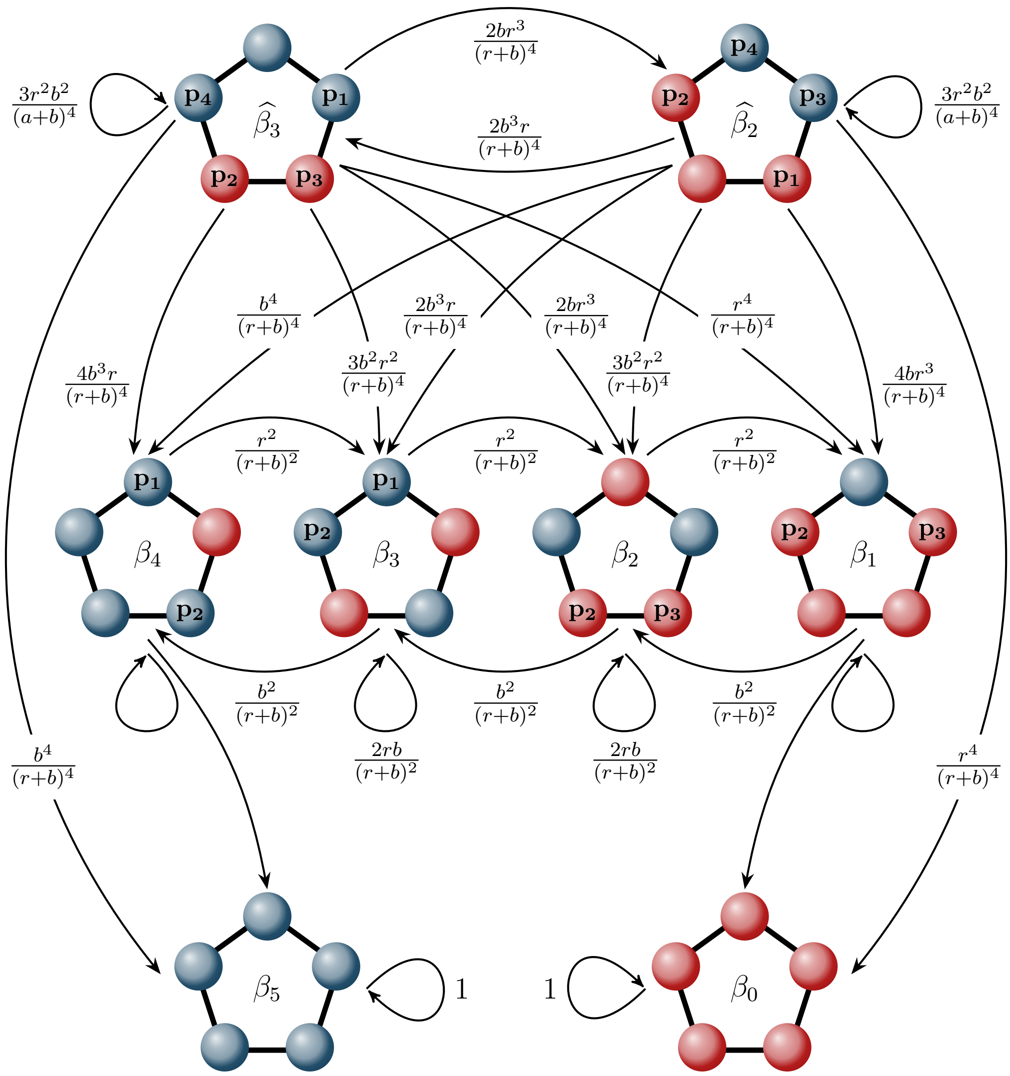


Figure 5.5: Possible States Up To Symmetry and Their Transition Probabilities of a Biased Consensus Game on C_5.

Proof. Here we exclude the consensus configurations because there are no coloured random walks to help us reconstruct S. Otherwise, take v_i and assume **wlog** that $f(S))(v_i) = (k, \text{red})$ for some $k \in [0,1]$. We can reconstruct S as follows: we know, by definition of f that $S(v_{i+1}) = \text{red}$. Then, $S(v_{i+3}) = \text{red}$ if and only if $R(v_{i+2}) = -1$. We apply this reasoning successively until S is fully determined. Note that when n is even, we can determine the colours of nodes in a given partition if and only if there are random-walking particles on the other partition at a given time. ■

If function f would not register the colour of the random-walking particles, we would have a two-to-one function, where inverse configurations (with all colours swapped) would map to the same R for $b = r$. More generally, inverse configurations, each in a game with swapped biases compared to the other, would map to the same R. Here, swapped biases means that bias towards blue in one game coincides with the bias towards red in the other and vice-versa.

Note that the higher the bias b, the greater the probability that a particle moves away from a node coloured *blue* in the corresponding biased consensus game. Similarly, by our definition above, the probability of a randomising node v choosing *blue* is equal to the probability that the corresponding random particle moves away from v's blue neighbour.

We aim to show that the two games are somehow related. Having a clearer understanding of how this relationship can be established might allow us to transfer the conclusions we derive from the random-walking particles scenario to the consensus one. We will shortly prove that having analogous games running in parallel will express similar behaviour. For example, we will show that the expected time taken for all the particles to disappear is equal to the expected time taken for a consensus to be achieved.

We at this point reach an impasse in our analogy that needs to be addressed. Although the expected time to reach consensus and the time to annihiate all particles seem to be the same, we have no clear method by which to differentiate the *blue* consensus from the *red* consensus just by looking at the particles game. This difficulty becomes evident when we observe that, at any point in an annihilating particles game, including in its last rounds, the number of blue and red particles is the same.

Proposition 5.3.12 *Consider the initial state of a process of annihilating random-walking particles in a cycle R_0 such that $R_0 = f(S_0)$, where S_0 is the initial configuration of a game $(\mathring{\mathcal{F}}_\Delta, S_0)$. Then, considering the process $\{R_i\}_{i\geq 0}$*

(i) The number of particles always decreases;

(ii) Considering a cycle of the form $(v_i, v_{i+2}, v_{i+4}, \ldots, v_i)$, the colours of the particles in R alternate;

(iii) Thus, particles that meet and annihilate each other are always of different colours.

(iv) The number of particles is always even;

Proof. Item (i) follows immediately from definition 5.3.7, as no particles are ever created. The fact that the number of particles always decreases will be of use after we show the equivalence between the two types of games, because then we will conclude that the number of randomising nodes on a biased consensus game also always decreases.

We now show Item (ii). Assume, **wlog**, that there is a blue particle in v_i, then, $v_{i+1} = \textit{blue}$ by definition. Also, $v_{i+3} = \textit{blue}$ if and only if there are no particles in v_{i+2}. Until no particles are present on nodes of the form v_{i+2j}, $j \in \mathbb{N}$, $v_{i+3} = \textit{blue}$. Thus, once a particle is present, its neighbouring clockwise node has to be red. Item (iii) now follows immediately. Item (iv) can be shown from Item (ii): if the colours of particles alternate in a given cycle, then there are an even number of such particles, for the same reason as an odd cycle is not two-colourable. ∎

We will now prove what evidence has been pointing at thus far: two corresponding games running in parallel and taking analogous random decisions, will still be corresponding games during all subsequent rounds.

Lemma 5.3.13 (Equivalence Between Games) *Let $\mathcal{S} \setminus \{\gamma_{blue}, \gamma_{red}\}$ be the state-space (excluding consensus states) for a given set of rules of biased consensus games $\mathcal{F}_\Delta$, and let $\mathcal{S} \setminus \{r_{-1}\}$ be the state-space (excluding no-particles state r_{-1}) for the set of rules of an annihilating random-walking particles game generated by applying the function f to states in $\mathcal{S}$.*

In these conditions, both digraphs that represent the states and transition probabilities of both games are isomorphic, with isomorphism f.

Proof. In order to prove that the two games are equivalent, we are going to show that the transition matrix of both Markov chains are equal, i.e., that the directed graphs H and $\widehat{H}$ that represent the Markov chains of the consensus game and the random-walking particles game, respectively, are isomorphic.

In order to show this isomorphism, we are going to restrict our analysis to n odd and the bijective function f restricted to $\mathcal{S} \setminus \{\gamma_{\text{blue}}, \gamma_{\text{red}}\}$. The extension from odd n to natural n is simple given that partitions act independently, as seen several times so far.

Let s_1 and s_2 be arbitrary states of H. We will show that the transition probability from s_1 to s_2 (i.e., the weight of the edge connecting the two states in H) is the same as the one from $r_1 = f(s_1)$ to $r_2 = f(s_2)$ in $\widehat{H}$.

Case 1: $Pr(S_{t+1} = s_2 \mid S_t = s_1) = 0$. We are going to show that, in this case, $Pr(R_{t+1} = r_2 \mid R_t = r_1) = 0$. Indeed, if s_2 cannot be reached by s_1 in one step, there must be a non-randomising node that behaves differently from what is expected. Let v_{i+1} be that node and say, **wlog**, that $S_t(v_i) = S_t(v_{i+2}) =$ blue (and thus $S_{t+1}(v_{i+1}) =$ red). Now there are two options: either v_{i-1} is randomising in S_i (and thus $R_t(v_{i-1}) \neq -1$, or v_{i-1} is not randomising in S_i (and thus $R_t(v_{i-1}) = -1$).

In the latter case, assuming v_{i-1} behaves as expected, we can conclude that $S_{t+1}(v_{i-1}) =$ blue. However, if that were the case, it would mean that there must be a particle on v_i in R_{t+1}. We would then conclude that r_1 cannot transition to r_2 since none of $v_i's$ neighbours host a particle in R_t. The only case in which this would hold is when v_{i-1} does not behave as expected, but then by induction we would end up either eventually reaching a randomising node or the entire graph would have only non-randomising nodes , which cannot be the case according to our hypothesis.

In the former case, if $R_t(v_{i-1}) \neq -1$, then $R_t(v_{i-1}) = \left(\frac{r}{b+r}, \text{blue}\right)$ has to be a blue particle because $S_t(v_i) =$ blue. Assume by contradiction that r_1 can lead to r_2 in a subsequent round. Then, if $S_{t+1}(v_{i-1}) =$ blue, then there must be a particle in v_i on R_{t+1} that must have come from v_{i-1}, thus it must be blue; however, $S_{t+1}(v_{i+1}) =$ red. Otherwise, if $S_{t+1}(v_{i-1}) =$ red, the probabilities involved are contradictory. In the case of different biases ($r \neq b$), $Pr\left(S_{t+1}(v_{i-1}) = \text{red}\right) = \frac{r}{b+r}$, which is correlated to the random particle in v_{i-1} on R_t deciding on walking clockwise, which we already established above to be impossible. However, the argument is not yet complete because we need to consider the unbiased case. Considering that the (blue) particle in v_{i-1} on R_t moves counter-clockwise, the only way we do not reach a contradiction (note that $S_{t+1}(v_{i-1} = \text{red})$) is if it gets annihilated by another particle coming from v_{i-3}. Note that this node, v_{i-3}, would have to be turning red from round t to $t+1$ as we know v_{i-2} is not randomising in S_{t+1} because particles just got annihilated by moving there. Finally, we observe that node v_{i-5} has to be randomising in S_t, otherwise it will be blue in S_{t+1}, leaving v_{i-4} to be randomising in S_{t+1} with no possibility of a particle having landed there. For the exact same reason as for v_{i-1}, we need $S_{t+1}(v_{i-5}) = \text{red}$. By induction, we see that the only way to avoid a contradiction is to have *all*[vi] nodes randomising and changing to red in S_{t+1}. However, by our hypothesis, $s_2 \neq \gamma_{\text{red}}$.

Case 2: $Pr(S_{i+1} = s_2 \mid S_i = s_1) = K > 0$. We are going to show that we also have $Pr(R_{t+1} = r_2 \mid R_t = r_1) = K$. We can calculate K by looking at the decision of randomising nodes from s_1 to s_2. Let, **wlog**, v_i be a randomising node in S_t with $S_t(v_{i+1}) = \text{red}$. Thus, $R_t(v_{i-1}) = \left(\frac{b}{r+b}, \text{red}\right)$. With probability $\frac{b}{r+b}$, $S_{t+1}(v_i) = \text{blue}$. In this case, v_{i+1} is randomising in S_{t+1} unless v_{i+2} was also randomising in S_t and chose blue. Note that, considering that particles walk towards its red neighbour with probability $\frac{b}{r+b}$ and towards its blue neighbour with probability $\frac{r}{r+b}$, we know that with the same probability that v_i chooses blue in S_t, the red particle in v_i in R_t decides to move clockwise, to be annihilated if and only if there was a (blue) particle in v_{i+2} that moved counter-clockwise (with probability $\frac{b}{r+b}$). Thus, the probability of v_{i+1} being randomising on S_{t+1} is the same as the probability of there being a particle in v_{i+1} on R_{t+1}. In conclusion, because the probabilities involved coincide for all nodes in C_n, $Pr(R_{t+1} = r_2 \mid R_t = r_1) = K$. ■

Note that the games having the same behaviour implies, in particular, that the expected time for the process to finish are also equal. In order to visualise this fact, consider two (correspondent) initial configurations $(\mathring{\mathcal{F}}_\Delta, S_0)$ and $R_0 = f(S_0)$ according to Definition 5.3.9. Each one of these two configurations lie on correspondent states of isomorphic Markov chains, and thus will have equal probabilities of reaching the absorbing states. Note that the pre-image of absorbing state in the random walks process is the set of (two) absorbing states in game $(\mathring{\mathcal{F}}_\Delta, S_0)$, i.e., $f^{-1}(r_{-1}) = \{\gamma_{\text{blue}}, \gamma_{\text{red}}\}$.

[vi] In case of even cycles, we would have all nodes in a given partition (instead of in the entire graph) to be changing to red.

5.3.1.2 Solving Biased Games on Cycles with Two Colours

When we consider annihilating random-walks that are moving between neighbouring nodes of a graph in a synchronous fashion, particles on neighbouring nodes at a given time might actually be far from meeting each other. That fact is particularly interesting in a cycle: the actual minimum distance (the number of steps necessary for an encounter) between random walks in neighbouring nodes in a cycle C_n in a given time is $n-1$.

The considerations described above motivate the definition that follows, in which we duplicate cycles in order to capture a more realistic distance between the particles in these games. We also apply an edge-colouring procedure to keep track of which of the two consensus have been achieved after all particles have disappeared.

Definition 5.3.14 (Duplication) *Let X be a set of colours and $\mathcal{S}(C_n)$ a set of colourings of a cycle C_n, $n \in \mathbb{N}$. We define the function $\boldsymbol{double}_{C_n}(s, x)$ that receives $s \in \mathcal{S}(C_n)$ and $x \in X$ as input creates a configuration $s' \in \mathcal{S}(C_{2n})$, i.e., $s' : C_{2n} \to X$, and such that*

$$s'(v_i) = \begin{cases} x & \text{if } i \text{ is odd,} \\ s(v_i) & \text{if } i \text{ is even and } i < n, \\ s(v_{i-n}) & \text{if } i \text{ is even and } i > n. \end{cases} \tag{5.12}$$

If $(\mathring{\mathcal{F}}_\Delta, S_0)$ is a biased two-colour consensus game on C_n, $n \in \mathbb{N}$, we say that the game $(\mathring{\mathcal{F}}_\Delta, S_0')$, played on C_{2n}, is the augmented *version of $(\mathring{\mathcal{F}}_\Delta, S_0)$. Finally, we colour the edges between two nodes coloured x with colour x, and the edges between nodes of different colour, say x and $\widetilde{x}$, with colour $\widetilde{x}$ (note we can never have two consecutive nodes coloured both $\widetilde{x}$ for $\widetilde{x} \neq x$). That will help us to keep track of the amount of blue and red nodes in the original game.*

Recall that a similar approach regarding colouring of edges has been presented in Chapter 3, in Definition 3.2.11. Also from Chapter 3, note that the duplication function has similarities with the split function in Definition 3.2.24.

Example 5.3.15. Consider Figure 5.6 for an illustration of the duplication function applied to a configuration of a game played in C_7. Note that $(\mathring{\mathcal{F}}_\Delta, S_0')$, on C_{14} is the *augmented* version of $(\mathring{\mathcal{F}}_\Delta, S_0)$, on C_7. For convenience, random walks are also in the same graph (represented by $p_1, \ldots p_4$): the odd indexes represent red particles, and even indexes represent the blue ones. Note that edges in $(\mathring{\mathcal{F}}_\Delta, S_0')$ are red if, and only if, they are connected to a red node. As usual, we denote the top most node

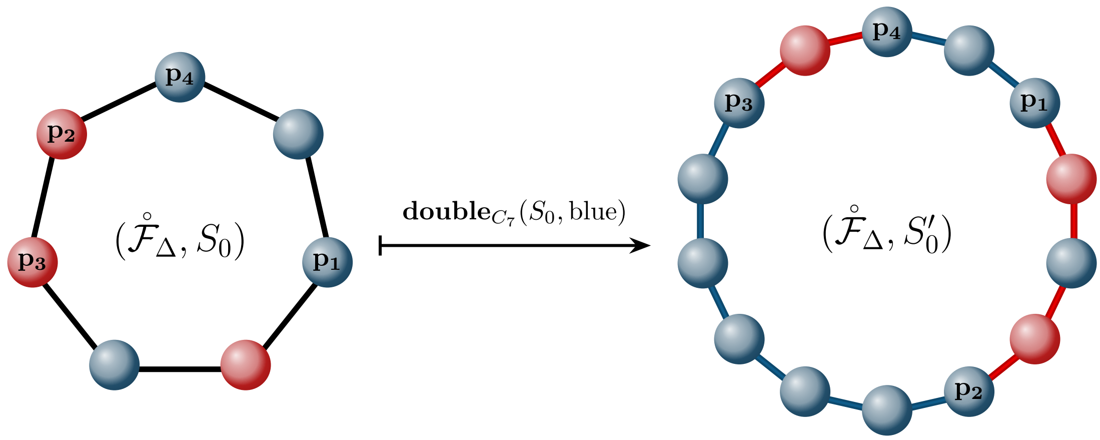


Figure 5.6: Application of $\mathbf{double}_{C_7}(S_0, \text{blue})$. Randomising nodes have particles on them, represented by p_1, p_2, p_3, and p_4.

in each graph as v_1, and move clockwise until v_7 in $(\mathring{\mathcal{F}}_\Delta, S_0)$, and v_{14} in $(\mathring{\mathcal{F}}_\Delta, S_0')$. Note that the blue particle p_4 sits on v_1 in both $(\mathring{\mathcal{F}}_\Delta, S_0)$ and $(\mathring{\mathcal{F}}_\Delta, S_0')$. The also blue particle p_2 sits on v_7 (node coloured red) in game $(\mathring{\mathcal{F}}_\Delta, S_0)$ and also on v_7 in game $(\mathring{\mathcal{F}}_\Delta, S_0')$. On the other hand, the red particle p_3 sits on v_6 in game $(\mathring{\mathcal{F}}_\Delta, S_0)$ and on v_{13} in game $(\mathring{\mathcal{F}}_\Delta, S_0')$.

We are now going to present properties of augmented games in order to better understand their behaviours and the connections with processes of annihilating random walks on cycles.

Remark 5.3.16. To improve readability, we are, for the rest of this chapter, going to assume colour x used as argument of the duplication function is the colour blue. Therefore, when we refer to the *augmented* version of a biased consensus game $(\mathring{\mathcal{F}}_\Delta, S_0)$, on C_n, we refer to game $(\mathring{\mathcal{F}}_\Delta, S_0')$, where $S_0' = \mathbf{double}_{C_n}(S_0, \text{blue})$.

Proposition 5.3.17 *Let $(\mathring{\mathcal{F}}_\Delta, S_0')$, on C_{2n} be the* augmented *version of a biased consensus game $(\mathring{\mathcal{F}}_\Delta, S_0)$, on C_n. Then, considering $(\mathring{\mathcal{F}}_\Delta, S_0')$,*

(i) Random walks are always positioned in odd nodes during even rounds, and even nodes during odd rounds (therefore always on blue nodes).

(ii) Between two consecutive random walks there are either only blue nodes, or an alternating sequence of red and blue nodes.

Proof. The proposition follows immediately from observing that no even node can be randomising in odd rounds, and no odd node can be randomising in even rounds. Also, a pair of consecutive particles defines a group of edges between them. These edges have to be of the same colour and the size of this group might either increase by 2, decrease by 2, or stay the same. ■

The previous definition can only be of use in a case in which there is some straightforward way to 'read' the results on the original game by looking only at the augmented game. The following proposition presents a solution for the problem of comparing the behaviours of the two types of games.

Proposition 5.3.18 *Let* $(\mathring{\mathcal{F}}_\Delta, S_0')$*, on* C_{2n} *be the* augmented *version of a biased consensus game* $(\mathring{\mathcal{F}}_\Delta, S_0)$*, on* C_n*. Then,*

(i) Blue consensus in $(\mathring{\mathcal{F}}_\Delta, S_0)$ *is represented by blue consensus in* $(\mathring{\mathcal{F}}_\Delta, S_0')$*.*

(ii) Red consensus in $(\mathring{\mathcal{F}}_\Delta, S_0)$ *is represented by failure to reach consensus in* $(\mathring{\mathcal{F}}_\Delta, S_0')$*, i.e., by reaching a losing state.*

(iii) The number of blue edges in $(\mathring{\mathcal{F}}_\Delta, S_0')$ *is twice the number of blue nodes in* $(\mathring{\mathcal{F}}_\Delta, S_0)$*.*

(iv) The number of red edges in $(\mathring{\mathcal{F}}_\Delta, S_0')$ *is twice the number of red nodes in* $(\mathring{\mathcal{F}}_\Delta, S_0)$*.*

Proof. The first step to prove all items above is to observe that the function **double** is objective when we consider only the partition that is not monochromatic at a given time. With that in mind, Items (i) and (ii) become immediate. To see the validity of Items (iii) and (iv), it is enough to observe that since one partition is monochromatically blue, each node in the other partition determines whether we have two red edges (if node is red) or two blue edges (if node is blue). ■

The following proposition gives us a way to control the number of blue (or red) nodes in a given game by looking at the colour of edges in the augmented version of that game. The main difficulty with non-augmented games is the fact that non-randomising nodes may or may not change according to their neighbours, so looking at the random decisions does not immediately give us a way to infer the number of a given colour in subsequent rounds. In other words, non-randomising nodes in augmented games always choose blue (regardless of their current colour).

Proposition 5.3.19 *Let* $(\mathring{\mathcal{F}}_{\Delta}, S_0')$, *on* C_{2n} *be the* augmented *version of a biased consensus game* $(\mathring{\mathcal{F}}_{\Delta}, S_0)$, *on* C_n. *Then, a given set of consecutive blue (resp. red) edges in a given round may either*

(i) Decrease by 2 *with probability* $\frac{r^2}{(r+b)^2}$ *(resp.* $\frac{b^2}{(r+b)^2}$*); or*

(ii) Increase by 2 *with probability* $\frac{b^2}{(r+b)^2}$ *(resp.* $\frac{r^2}{(r+b)^2}$*); or*

(iii) Stay unchanged with probability $\frac{2br}{(r+b)^2}$ *(resp.* $\frac{2br}{(r+b)^2}$*).*

Note that considering only one colour, the probabilities of growth of different sets are independent, which is not the case if we consider any pair of neighbouring sets (of consecutive edges) at the same time.

Proof. It is enough to note that a set of consecutive edges of the same colour in augmented games have always an even number of edges. In other words, random-walking particles cannot 'jump over each other'. In other to confirm probabilities shown in Items (i), (ii), and (iii), we simply use the fact that a blue (resp. red) particle has a probability of $\frac{b}{r+b}$ (resp. $\frac{r}{r+b}$) to *move away* from the nearest blue (resp. red) edge. together with the fact that the change on size of each set of consecutive edges depends on the movement of each of two particle at both its ends. Neighbouring sets of consecutive edges do not have independent growth as they share a common particle, which behaviour affect both sets. ■

We are about to answer Question **E2** by exploring Question **E3**: we define a random variable and prove it is a martingale with respect to the configuration of the augmented version of a given game in C_n.

Definition 5.3.20 *Let* $(\mathring{\mathcal{F}}_{\Delta}, S_0)$ *be a biased two-colour consensus game on* C_n, $n \in \mathbb{N}$. *Also let* k_t *be the number of blue nodes in* S_t, *and the function* $\boldsymbol{B}(k_t) := \sum_{j=1}^{k_t} b^{2(n-1-j)} r^{2(j-1)}$ *be a polynomial defined by the integer* k_t. *Finally, we define the random variable* Y_t *as*

$$Y_t := \boldsymbol{B}(k_t) = \sum_{j=1}^{k_t} b^{2(n-1-j)} r^{2(j-1)} \tag{5.13}$$

Our current conjecture is that, for game $(\mathring{\mathcal{F}}_{\Delta}, S_0)$ on C_n the random variable given by Y_t is a martingale with respect to S_t. To prove this, we are going to focus our attention on the augmented version $(\mathring{\mathcal{F}}_{\Delta}, S_0')$ on C_{2n}. The following example illustrates the steps of the proof of Theorem 5.3.22, to be presented shortly.

EXAMPLE 5.3.21 (SOLVING A SIMPLE CASE). Consider the game $(\mathring{\mathcal{F}}_\Delta, S_0')$, on C_7 described in Figure 5.6 and its augmented version. We are going to show that Y_t is a martingale for this game. Note that

$$Y_0 = \mathbf{B}\,(4) = b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6 \tag{5.14}$$

We will show that $\mathbb{E}(Y_1) = Y_0$. Indeed,

$$\begin{aligned}
\mathbb{E}(Y_1) &= \frac{1}{(r+b)^4}\left[b^4B(6) + 4b^3rB(5) + 6b^2r^2B(4) + 4br^3B(3) + r^4B(2)\right] = \\
&= \frac{1}{(r+b)^4}[b^4(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6 + b^4r^8 + b^2r^{10}) + \\
&\qquad +4b^3r(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6 + b^4r^8) + \\
&\qquad +6b^2r^2(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6) + \\
&\qquad +4br^3(b^{12} + b^{10}r^2 + b^8r^4) + \\
&\qquad +r^4(b^{12} + b^{10}r^2)] \\
&= \frac{1}{(r+b)^4}[b^4(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6) + r^4(b^8r^4 + b^6r^6) + \\
&\qquad +4b^3r(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6) + 4br^3(b^6r^6) + \\
&\qquad +6b^2r^2(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6) + \\
&\qquad +4br^3(b^{12} + b^{10}r^2 + b^8r^4) + \\
&\qquad +r^4(b^{12} + b^{10}r^2)]
\end{aligned}$$

which gives us

$$\mathbb{E}(Y_1) = \frac{1}{(r+b)^4}\left[(r+b)^4(b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6)\right] = Y_0$$

The fact that Y_t is a martingale will be formally proven in Theorem 5.3.22. We will then show that the probability in this case is given by

$$Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6}{b^{12} + b^{10}r^2 + b^8r^4 + b^6r^6 + b^4r^8 + b^2r^{10} + r^{12}} \tag{5.15}$$

Theorem 5.3.22 (Probability of Consensus in Biased Games on Cycles) *Let $(\mathring{\mathcal{F}}_\Delta, S_0)$ be a biased two-colour consensus game on C_n, n odd, and let i be the number of blue nodes in S_0. Then,*

$$Pr(S_\tau = \gamma_{blue} \mid S_0) = \frac{\sum_{j=1}^{i} b^{2(n-1-j)}r^{2(j-1)}}{\sum_{j=1}^{n} b^{2(n-1-j)}r^{2(j-1)}} = \frac{1-\left(\frac{r}{b}\right)^{2i}}{1-\left(\frac{r}{b}\right)^{2n}} \tag{5.16}$$

We can get the probability of consensus in red using an analogous formula in which i represents the number of red nodes in S_0.

Proof. Using Lemma 5.3.13 and Property 5.3.18, is enough to show that the result is also the probability of the analogous random-walk process of the augmented version $(\mathring{\mathcal{F}}_\Delta, S'_0)$ on C_{2n}. As usual, S'_t stands for the random variable representing the configuration of the augmented game in round t. We show that the random variable $Y_t = B(i) = \sum_{j=1}^{i} b^{2(n-1-j)} r^{2(j-1)}$, where i is **half** the quantity of blue edges[vii] in S'_t, is a martigale with respect to S_t. Note that each set of blue edges in a given round changes according to probabilities given in Proposition 5.3.19. Note also that possible annihilations due to encounter of two random walks, and therefore connection of blue or red sets of edges, do not change the sum of blue nor red edges, but solely the number of random-walking particles from one round to another.

Let w_t be the number of random-walking particles in round t. Note that w_t is always less than or equal to the number of blue edges. We then have

$$\begin{aligned}\mathbb{E}(Y_{t+1} \mid S_t) &= \frac{1}{(r+b)^{w_t}} \left[\mathbf{B}\left(i + \frac{w_t}{2}\right) b^{w_t} + \binom{w_t}{1} \mathbf{B}\left(i + \frac{w_t - 2}{2}\right) b^{w_t - 1} r + \right. \\ &\qquad \left. \cdots + \mathbf{B}\left(i - \frac{w_t}{2}\right) r^{w_t} \right] \qquad (5.17) \\ &= \frac{1}{(r+b)^{w_t}} \sum_{j=0}^{w_t} \binom{w_t}{j} \mathbf{B}\left(i + \frac{w_t - 2j}{2}\right) b^{w_t - j} r^j. \qquad (5.18)\end{aligned}$$

In order to simplify notation, we define $L = \frac{w_t - 2k}{2}$. The main step in this proof is to use that $\binom{w_t}{j} = \binom{w_t}{w_t - j}$ in order to prove that

$$\binom{w_t}{k} \mathbf{B}(i + L) b^{w_t - k} r^k + \binom{w_t}{w_t - k} \mathbf{B}(i - L) b^{w_t - (w_t - k)} r^{w_t - k} \qquad (5.19)$$

is equal to

$$\binom{w_t}{k} \mathbf{B}(i) \left(b^{w_t - k} r^k + b^k r^{w_t - k}\right)$$

For an integer k. We assume, **wlog**, $k < \frac{w_t}{2}$. Indeed, developing Equation 5.19 we get

$$\begin{aligned}(5.19) &= \binom{w_t}{k} \left[\mathbf{B}(i + L) b^{w_t - k} r^k + \mathbf{B}(i - L) b^k r^{w_t - k}\right] \\ &= \binom{w_t}{k} \left[\mathbf{B}(i) \left(b^{w_t - k} r^k + b^k r^{w_t - k}\right) + K\right]\end{aligned}$$

where

$$K = \left(\sum_{j=i+1}^{i+L} b^{2(n-1-j)} r^{2(j-1)}\right) b^{w_t - k} r^k - \left(\sum_{j=i-L+1}^{i} b^{2(n-1-j)} r^{2(j-1)}\right) b^k r^{w_t - k} = 0$$

[vii]Note that i is also the number of blue nodes in S_t.

This can be seen by applying the substitution $s = j - \frac{-w_t+2k}{2} = j + L$ on the second sum, getting

$$\begin{aligned} K &= \left(\sum_{j=i+1}^{i+L} b^{2(n-1-j)} r^{2(j-1)} \right) b^{w_t-k} r^k - \left(\sum_{s=i+1}^{i+L} b^{2(n-1-s+L)} r^{2(s-L-1)} \right) b^k r^{w_t-k} \\ &= \left(\sum_{j=i+1}^{i+L} b^{2(n-1-j)+w_t-k} r^{2(j-1)+k} \right) - \left(\sum_{s=i+1}^{i+L} b^{2(n-1-s)+w_t-k} r^{2(s-1)+k} \right) = 0 \end{aligned}$$

Going back to Equation 5.17, we have

$$\begin{aligned} \mathbb{E}(Y_{t+1} \mid S_t) &= \frac{1}{(r+b)^{w_t}} \sum_{j=0}^{w_t} \binom{w_t}{j} \mathbf{B}\left(i + \frac{w_t - 2j}{2}\right) b^{w_t-j} r^j \\ &= \frac{1}{(r+b)^{w_t}} \left[\binom{w_t}{\frac{w_t}{2}} \mathbf{B}(i)\, b^{\frac{w_t}{2}} r^{\frac{w_t}{2}} + \sum_{j=0}^{\frac{w_t}{2}-1} \binom{w_t}{k} \mathbf{B}(i) \left(b^{w_t-k} r^k + b^k r^{w_t-k}\right) \right] \\ &= \mathbf{B}(i) \end{aligned}$$

Finally, since b and r are constants and $\mathbf{B}(i)$ is bounded (by 0 and $\mathbf{B}(n)$), and considering the game ends at round τ (random walks meet eventually with probability 1), we apply Doob's Stopping Theorem to get

$$Y_0 = \mathbb{E}(Y_0) = \mathbb{E}(Y_\infty) = \mathbb{E}(Y_\tau) = \mathbf{B}(0)\, Pr(S_\tau \neq \gamma_{\text{blue}} \mid S_0) + \mathbf{B}(n)\, Pr(S_\tau = \gamma_{\text{blue}} \mid S_0)$$

Thus,

$$Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{\mathbf{B}(i)}{\mathbf{B}(n)} \tag{5.20}$$

■

Corollary 5.3.23 *We extend the results for a general cycle C_n. Let k_1 (resp. k_2) be the number of blue nodes in partition 1 (resp. 2). Then,*

$$Pr(S_\tau = \gamma_{blue} \mid S_0) = \frac{\boldsymbol{B}(k_1) \times \boldsymbol{B}(k_2)}{\left(\boldsymbol{B}\left(\frac{n}{2}\right)\right)^2} = \frac{\left(1 - \left(\frac{r}{b}\right)^{2i}\right)\left(1 - \left(\frac{r}{b}\right)^{2i}\right)}{1 - 2\left(\frac{r}{b}\right)^{2n} + \left(\frac{r}{b}\right)^{4n}} \tag{5.21}$$

Analogously, we have

$$Pr(S_\tau = \gamma_{red} \mid S_0) = \frac{\boldsymbol{B}\left(\frac{n}{2} - k_1\right) \times \boldsymbol{B}\left(\frac{n}{2} - k_2\right)}{\left(\boldsymbol{B}\left(\frac{n}{2}\right)\right)^2}. \tag{5.22}$$

Finally, the probability of non-convergence is given by

$$Pr(S_\tau \notin \Gamma \mid S_0) = \frac{\boldsymbol{B}\left(\frac{n}{2} - k_1\right) \times \boldsymbol{B}(k_2) + \boldsymbol{B}(k_1) \times \boldsymbol{B}\left(\frac{n}{2} - k_2\right)}{\left(\boldsymbol{B}\left(\frac{n}{2}\right)\right)^2}. \tag{5.23}$$

Proof. We just combine Theorems 5.3.22 and 3.2.26. ■

We can now conclude that the answer to both Questions **E2** and **E3** is **yes**: the relative position of particles of the same colour in an odd cycle (or within the same partition in an even cycle) is irrelevant for the probability of convergence to each of the consensus states, which comes from the fact that there is a martingale that only takes in account the number of nodes of each colour, as proven by Theorem 5.3.22. We now apply the results above to the motivational problem posed in the beginning of this chapter.

SOLUTION TO PROBLEM 5. Going back to the game depicted in Figure 5.1, we can calculate the probabilities of each of the two colours winning. Applying Theorem 5.3.22, we get

$$Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{b^{32} + b^{30}r^2 + \cdots + b^{20}r^{12} + b^{18}r^{14}}{b^{32} + b^{30}r^2 + \cdots + b^2r^{30} + r^{32}} \tag{5.24}$$

We have $\delta_{\text{blue}} = 2$ and $\delta_{\text{red}} = 1$, thus

$$Pr(S_\tau = \gamma_{\text{blue}} \mid S_0) = \frac{1 - \left(\frac{1}{2}\right)^{16}}{1 - \left(\frac{1}{2}\right)^{34}} \approx 0.99998 \tag{5.25}$$

5.4 Interesting Ramifications

We now motivate and formally define three family of problems that derive from the study of biased consensus games. For the reachability problem (Section 5.4.1), we provide a full solution for cycles based on the correspondence with random walks studied earlier in this chapter.

5.4.1 The Reachability Problem

Consider a biased consensus game $(\mathring{\mathcal{F}}_\Delta, S_0)$ that starts as in Figure 5.7 (left). Is there a positive probability that it will eventually reach $s \in \mathcal{S}$ as in Figure 5.7 (right)? We reproduce here a formal definition of this problem from [1].

Definition 5.4.1 (Markov Reachability Problem) *Given a finite stochastic matrix $\boldsymbol{M}$ with rational entries and given r rational, does there exist $t \in \mathbb{N}$ such that $(\boldsymbol{M}^t)_{1,2} > r$?*

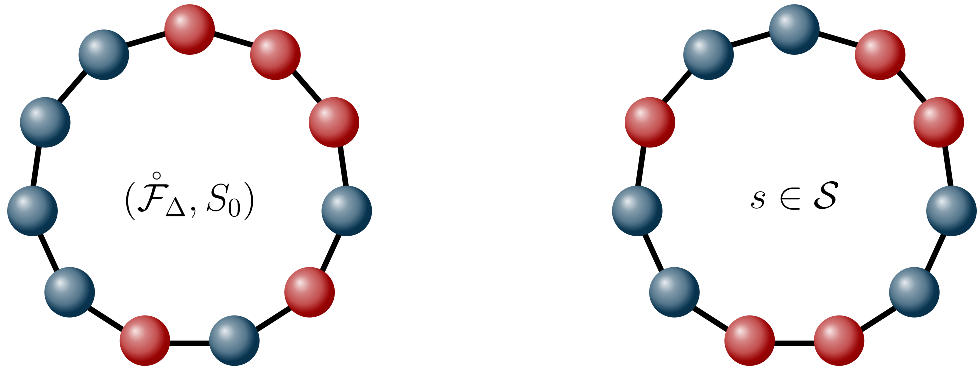


Figure 5.7: The Initial Configuration of a Game $(\mathring{\mathcal{F}}_\Delta, S_0)$ and a Given Configuration $s \in \mathcal{S}$. Can $(\mathring{\mathcal{F}}_\Delta, S_0)$ Eventually Reach s?

There is evidence that this is a hard problem in general as there is a reduction to it from the *Positivity Problem* [1]. In turn, there is a reduction to the Positivity Problem from the *Skolem Problem* [58]. Finally, Skolem Problem is known to be NP-Hard [8].[viii] We look at the particular case in which $\mathbf{M}$ describes a consensus game on an undirected cycle C_n, and $r = 0$. Also, for states $s, \widetilde{s} \in \mathcal{S}$, we want to know whether there exists $t \geq 0$ such that $(\mathbf{M}^t)_{s,\widetilde{s}} > 0$, i.e., whether there is a positive probability that a process starting at state s reaches state $\widetilde{s}$ in t steps.

To provide a visual insight for the next theorem, it is helpful to refer back to Figure 5.5, where random-walking particles are depicted in each configuration of a game on C_5. Note that in each level the number of particles decreases and that there is no possibility for a game to 'move up a level'. Note also that each configuration in a given level reaches all the other ones from that same level (and consequentially all the ones below as well). We now present the formal theorem and its proof.

Theorem 5.4.2 (Reachability Problem for Cycles) *Let $(\mathring{\mathcal{F}}_\Delta, S_0)$, with $S_0 \notin \Gamma$, be a biased consensus game on a cycle C_n with two colours, and a state $s \in \mathcal{S}$, where $\mathcal{S}$ denotes, as usual, the set of all colourings of C_n with two colours.*

(i) If n is odd, let N and M be the numbers of randomising nodes[ix] *in S_0 and s,*

[viii] Although we are not going to use these definitions any further, here we provide, for completeness, a brief description of the Positivity Problem and the Skolem problem. The former can be understood as the decision problem in which, given a linear recurrence relation, we want to know whether all its terms are positive. The latter is the decision problem in which, given a recurrence relation, we want to know whether if ever reaches the value zero.

[ix] Here we abuse notation since the colouring s is not part of a process, therefore the term 'randomising nodes' should be understood as 'nodes between two neighbours of different colour' .

respectively. In these conditions,

$$\exists t \geq 0\,[Pr(S_t = s \mid S_0) > 0] \Longleftrightarrow N \geq M \tag{5.26}$$

(ii) If n is even, let N_1 and N_2 be the numbers of randomising nodes in each of the partitions of C_n in S_0. Analogously, let M_1 and M_2 be the number of randomising nodes in each of the partitions of C_n in s_0. We assume, ***wlog****, that $N_1 \leq N_2$ and that $M_1 \leq M_2$. In these conditions, if $N_1, N_2 > 0$, then*

$$\exists t \geq 0\,[Pr(S_t = s \mid S_0) > 0] \Longleftrightarrow [N_1 \geq M_1] \wedge [N_2 \geq M_2] \tag{5.27}$$

Else, if $N_1 = N_2 = 0$, then S_0 is a losing game (since $S_0 \notin \Gamma$), and can only reach s if s is also a losing game. If $N_2 > N_1 = 0$, then S_0 has a γ-monochromatic partition. And thus will only reach s if and only if s also has a γ-monochomatic partition (not necessarily the same partition) for the same $\gamma \in \Gamma$, and $N_2 \geq M_2$.

Proof. Part (i) comes immediately by observing that there is a bijective function (Proposition 5.3.11) from biased consensus games to process of annihilating random-walking particles on the same odd cycle. Recall that the number of randomising nodes in the Flag Coordination Game corresponds (by Definition 5.3.9) to the number of particles in the random walks game. Because particles may jump over each other on the odd cycle (note that they move synchronously), any configuration of m particles can be reached from any other configuration of m particles.

Part (ii) can be shown by using the independence of partitions, and the fact that, in the presence of monochromatic partitions, they alternate from one round to the next. Note that you can get inverse configurations in the biased consensus game by having red particles move where blue particles were initially and vice-versa in the random walks process. ■

Going back to Figure 5.7, note that $Pr(S_t = s \mid S_0) = 0$, since $N = 4 < 10 = M$. We now provide a solution for the second part of Problem 1, from Chapter 2.

Solution to Problem 1 (Take 2). Recall that we considered a line of autonomous robots in a bucket brigade aiming to choose an action (colour) different from their neighbours', i.e., playing an anti-consensus game. Although we will not provide a proof in this dissertation, it is not hard to see that we can also have a correspondence between generalised consensus problems on paths and random-walking particles. Note that particles in the interior of a path behaves like in a cycle. If they hit either end of the path, however, they disappear. Note that they also disappear if they meet another particle, as in cycles. In these conditions, we can analyse whether configuration A is reachable by configuration B and vice versa. Using terminology from Theorem 5.4.2, we have, for configuration A, $N_1 = 1$ and $N_2 = 1$. For configuration B, $M_1 = 1$ and $M_2 = 4$. Thus, a game that starts as B has a positive probability of eventually reaching A, however, a game that starts in A can never reach B.

5.4.2 Trade-off Between Bias and Presence on the Graph

In order to motivate our exploration of Question **E6**, let us look into the solution of Problem 5 in more detail. We know that blue is far more likely to win than red. At this point, we might suspect that having only 7 instead of 8 nodes initially would be enough for blue to be more likely to win. That is indeed the case. This will be immediate once we show that just *one* blue node is what is needed to have a consensus in blue more likely than in red. The intuition is simple: $r = 1$ makes $\mathbf{B}(m)$ a sum of powers of two. And this sum does not reach 2^{m+1}. Therefore,

$$\frac{\mathbf{B}(1)}{\mathbf{B}(n)} > \frac{1}{2} \tag{5.28}$$

The solution of the above problem invites us to consider whether a slightly smaller bias for blue would still guarantee a greater likelihood to win even with one node in an initial configuration. For example, is one blue node still enough in case $r = 1$ and, say, $b = 1.9$? More formally, fixing $r = 1$, we are looking for

$$\inf_{b \in \mathbb{R}} \left\{ b \;\middle|\; \frac{\mathbf{B}(1)}{\mathbf{B}(n)} > \frac{1}{2} \right\} \tag{5.29}$$

We solve a particular case of this problem for C_3 in the following example.

EXAMPLE 5.4.3. Consider a biased consensus game on C_3. The threshold for a positive bias $\delta_{\text{blue}} = b$ for blue to be more likely to win starting from a configuration of one blue node and two red ones is given by

$$\inf_{b \in \mathbb{R}} \left\{ b \;\middle|\; \frac{\mathbf{B}(1)}{\mathbf{B}(n)} > \frac{1}{2} \right\} = \phi^{\frac{1}{2}} \tag{5.30}$$

where here ϕ denotes the golden ratio, i.e., solution for the equation $x^2 - x - 1 = 0$.

Motivated by the examples above, we define a problem to be investigated in future work. Solving it will allow us to understand what is the trade-off between the bias towards a given colour and how many nodes of that given colour there are in the network.

Definition 5.4.4 *Let $(\mathring{\mathcal{F}}_\Delta, S_0)$ be a biased two-colour consensus game on an odd cycle C_n as in Definition 5.3.4. Let $p \in [0,1]$ be the minimum probability of winning the game we want blue to have. We define*

$$\Omega(p,m) := \inf_{b\in\mathbb{R}} \left\{ b \;\middle|\; \frac{\boldsymbol{B}(m)}{\boldsymbol{B}(n)} > p, \;\; \textit{and } \delta_{blue} = b \right\} \tag{5.31}$$

as the lower threshold for a positive bias $\delta_{blue} = b$ for which a game starting with m blue nodes will have probability of blue winning greater than p. On the other hand, we define

$$\mho(p,b) := \min_{m\in\mathbb{N}} \left\{ m \;\middle|\; \frac{\boldsymbol{B}(m)}{\boldsymbol{B}(n)} > p, \;\; \textit{and } \delta_{blue} = b \right\} \tag{5.32}$$

as the minimum number of nodes that need to be blue in order for blue to have a probability higher than p assuming that the bias towards blue is $\delta_{blue} = b$.

5.4.3 Multiple Consecutive Biased Games

In this section we propose a problem to be investigated in future work. Consider a sequence of biased consensus games in which the bias towards colours change from one game to the other based on the previous consensus result. For example, assume an unbiased consensus game is to happen on a cycle with a given initial configuration S_0 with colours red and blue. The colour that wins the unbiased game will, say, have the bias towards it increased by 1 if already greater than 1, or decrease the opposing colour's bias by 1 otherwise. If blue wins, for example, the following game will be a biased game with $\delta_{\text{blue}} = 2$, and $\delta_{\text{red}} = 1$, starting at the same configuration S_0. We say that the entire process ends whenever either colour reaches a positive bias of a given integer $M > 0$.

This family of problems can be used to model path-dependent technologies, processes in which the very use of a technology partially acts as a self-fulfilling prophecy regarding the standard to be adopted, i.e., the more a given standard is used, the harder it is to adopt a different one. A standard example of path-dependent technology is the QWERTY keyboard.

We now provide a formal definition, framing this process as a random walk on a line.

Definition 5.4.5 (Multiple Consecutive Biased Games) *Consider the family of biased games $\{(\mathcal{F}_\Delta, S_0)_z\}_{z\in\mathbb{Z}}$ with set of biases varying according to z. More precisely, $\Delta(z) = \{\delta_{blue}(z), \delta_{red}(z)\}$. Consider now a (lazy) random walk*[x] *on the integer line which position is described by the random variable $\{Y_t\}_{t\in T}$ indexed on a discrete time-set and with the update rule given by*

$$Y_{t+1} = \begin{cases} Y_t + 1, & \text{with probability } Pr(S_\tau = \gamma_{blue} \mid (\mathcal{F}_\Delta, S_0)_{z=Y_t}) \\ Y_t - 1, & \text{with probability } Pr(S_\tau = \gamma_{red} \mid (\mathcal{F}_\Delta, S_0)_{z=Y_t}) \\ Y_t, & \text{otherwise} \end{cases} \tag{5.33}$$

We assume this random walk starts on $z = 0$ and gets absorbed on points $z = M$, and $z = -M$, for some constant $M > 0$.

Note that we allow the random walk defined above to be lazy so we can include games that might not converge (probability represented by the probability that the random walk does not move on that round).

In multiple consecutive biased games, we are interested in the probability of reaching either absorbing state given the function $\Delta(z)$, as well as the expected time for this process to end.

5.5 Summary of Results

In this chapter, we have studied a family of Flag Coordination Games in which there are different biases towards different colours or flags. Although we defined the problem in general, the focus of our attention was on the behaviours of such games on cycle graphs. As the first step of our analysis, we discarded the possibility that the probabilities of a given opinion to win increased linearly with the number of nodes initially coloured according to that opinion (Question **E1**). We established the probabilities to win of each colour by defining a martingale that describes the process, firstly in cycles of odd length (Theorem 5.3.22); then, we applied the results of Chapter 3 to extend the results for cycles of any length, resolving Question **E3** for cycles. Such probabilities do not depend on the initial relative position of nodes (in a given partition, if even cycle), but solely on the number of nodes of each colour that are present in the initial configuration of the game, answering Question **E2** for cycles.

As part of this process of solving Questions **E1**, **E2**, and **E3**, we formally addressed Question **E4** by showing a correspondence between a biased generalised

[x] A random walk is said to be *lazy* if there is a probability that it does not move at each round.

consensus game on the cycle and a process of annihilating random walks on the same cycle. This correspondence allowed us to determine whether a given state is reachable by a game starting at an arbitrary initial configuration (Question **E5**).

Finally, we introduced and formally defined two families of interesting ramifications of biased consensus games. In the first, we proposed a deeper study into the trade-off between biases in a consensus game and the number of nodes of each colour (Question **E6**). In particular, what is the lower bound for bias towards a given colour x that will allow x to be more likely to win than not even in situations in which only one node is initially coloured x? The second family of problems to be studied considers that multiple iterations of a game have been played in sequence, with the difference that biases may vary from one iteration to the next. With that, we may be able to model processes such as path-dependent technologies in which current or past consensus may affect future consensus.

Chapter 6

Conclusions and Future Work

In this thesis, we studied decentralised multi-agent processes with restricted communication capabilities, in which the only information each agent could send is their current state. In Chapter 2, we showed that there are a wide range of problems from different fields of study that can be viewed as Flag Coordination Games. For instance, problems in Graph Theory, such as proper colouring of graphs, can be framed as Flag Coordination Games (Example 2.2.7); Epistemic Logic, as portrayed in the Muddy Children Problem (Example 2.2.6); and Statistical Mechanics appeared as a Flag Coordination Game in Example 2.2.10. Later, we applied Flag Coordination Games to the Theory of Argumentation (Chapter 4) and saw that some random walk processes on graphs correspond to a family of consensus games (Chapter 5).

Not all sets of rules of Flag Coordination Games guarantee convergence for all possible initial configurations, and Chapter 3 fully solves that problem by providing a criterion for generalised consensus games in digraphs. Moreover, in Chapter 3 we presented probabilities for the convergence to each one of the goal states, given an initial configuration of the game, on any undirected or directed graph. Finally, Chapter 5 introduced the concept of biased (generalised) consensus games, in which nodes show a tendency to choose a particular colour (goal state) by having different weights associated to each colour. The problem is then fully solved for cycles when only two colours are involved.

6.1 Summary of Results

We now reproduce and then provide a more detailed account of each of the questions raised in Chapter 1.

A1 *Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the probability that the sequential decision process will enter an infinite cycle that does not converge to a pre-specified global goal state (i.e., an infinite cycle of non-convergence)?*

A2 *Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the probability that the sequential decision process will converge to a pre-specified desired global goal state?*

In Chapter 3 we studied a particular class of Flag Coordination Games in which the number of goal states is equal to the number of colours or flags available to agents in the game. Moreover, each agent is coloured differently in each goal state, which makes it possible for each agent together with its current state to be matched with one and only one goal state. Games under these conditions are called *generalised consensus games* (Definition 3.2.1).

Considering undirected graphs, generalised consensus games arise in different possible scenarios. One example of a situation in which agents of two clearly distinct groups make their decisions based on agents in the other group is a doctors and patients bipartite network in which colours refer to different health insurance providers. Alternatively, we can think of the Ising model for antiferromagnetism, in which, spins in one partition tend to be in the opposite direction to the ones in the other partition (Example 2.2.10). Note that these two examples represent two sides of the same coin: one is a consensus game and the other is anticonsensus. Both are particular cases of generalised consensus games. In the context of generalised consensus games on bipartite graphs, we derive a formula for the probability of convergence in each of the goal configurations as well as the probability of reaching a goal configuration at all, as opposed to entering an infinite loop (Theorem 3.2.26).

A3 *Given a defined set of rules of a Flag Coordination Game and given an initial state, what is the expected number of decision rounds (time steps) to reach a pre-specified global goal state?*

Also in Chapter 3, we computed formulas for lower and upper bounds on the expected number of rounds a game in an undirected bipartite graph would take, given its initial configuration, until reaching either a winning state or entering a loop (Theorems 3.2.28 and 3.2.19) in generalised consensus games with two colours and without bias. We used results on Zagreb indices [20] on graphs in order to provide a tighter lower bound for the process to end. We reproduce Table 3.1 in Table 6.1

	Single-partition games on bipartite graphs	General games on bipartite graphs
Winning probability $Pr(S_\tau = \gamma \mid S_0)$	$\dfrac{Y_0}{\lvert E\rvert}$	$\dfrac{Y_0 X_0}{\lvert E\rvert^2}$
Upper-bound for expected duration $\mathbb{E}(\tau)$	$mY_0 - Y_0^2$	$m(Y_0 + X_0) - (Y_0^2 + X_0^2)$
Lower-bound for expected duration $\mathbb{E}(\tau)$	$\dfrac{8(mY_0 - Y_0^2)}{mn} + 1$	$\dfrac{4(m(Y_0 + X_0) - (Y_0^2 + X_0^2)}{mn}$

Table 6.1: (Reproduce of Table 3.1) Summary of Results of This Chapter for Undirected Graphs.

A4 *Which sufficient conditions on the rules of a Flag Coordination Game are such that, for at least one possible initial state, there is a positive probability that the state loop described in Question **A1** is entered?*

To try to understand the necessary and sufficient conditions for losing configurations to appear in more general Flag Coordination Games, we explored what could lead to state loops of size other than 2 (as observed in bipartite graphs). In Chapter 3, we introduced the generalised consensus games on directed graphs and showed in Lemma 3.3.7 that losing loops of size k may appear if and only if the greatest common divisor of the length of all cycles in the graph is k. We have shown that the possibility of reaching losing configurations in generalised consensus games involve processes that become deterministic at some point (in loops, the interference of bias towards a given colour is irrelevant). Thus, this result is also valid for biased consensus games.

Moreover, we answered Questions **A1**, and **A2** for any digraph in Theorem 3.3.16.

A5 *How can we apply the concept of Flag Coordination Games to the field of Argumentation Theory to study a form of distributed argumentation in which each argument is controlled by an independent agent?*

In Chapter 4 we introduced the concept of Team Persuasion (Definition 4.3.2) as a Flag Coordination Game in which each agent controls one argument and has to decide whether or not to re-state it from time to time. There is a topic argument, which some agents are (directly or indirectly) attacking, and others are (directly or indirectly) defending. In this model, we only consider bipartite argumentation frameworks, and therefore we can separate all the arguments into two teams (defending and attacking). If an argument is stated, we say it is *on*. We consider that if the argument is not stated again when it could have been (there are differences between synchronous and asynchronous games), the audience forgets it and it is then considered to be *off*. We allow agents to have distinct algorithms that might include the assignment of weights on attacks, and more generally on any path that ends at the agent.

In these conditions, we studied the probability, given an initial configuration, of the topic argument to be accepted in the long run of this process, and the probability that it ends up being rejected (Corollary 4.4.10). For synchronous games, there is also a positive probability that the game enters a loop (related to Question **A4**). The only small difference between results in Chapter 4 and earlier ones in Chapter 3 is the introduction of a topic argument. In these situations, arguments that do not affect the topic neither directly nor indirectly have no influence in the final acceptability of the topic, and therefore are ignored when calculating probabilities of convergence to goal states. Remember that in weakly connected graphs we may have strongly connected components of the graph that cannot be either the start or endpoint of a path to or from the topic argument.

A6 *How can a Flag Coordination Game be influenced by external agents?*

The context of Team Persuasion Games is ideal to consider the influence of external agents, or *bribers*, who seek to locally modify the configuration of the game in order to achieve a given global state. More specifically, a briber would pay (in function of their payoff from the game's result) a given agent to change their argument from *off* to *on*, or vice versa. For the scenario with one briber, it is enough to calculate the change in probability of their team of choice winning for each possible change (Lemma 4.5.4). For two bribers working for opposing teams in synchronous games, however, they have to consider each others' possible choices when making their own. We then provide a game-theoretical analysis (Proposition 4.5.9) of this game on bipartite digraphs, which is essentially the same as the analysis for a general digraph. We also show that there is always at least one pure strategy Nash equilibrium for such games (Theorem 4.5.10).

A7 *What is the impact of the introduction of bias towards a given opinion (or flag colour) in the set of rules of a Flag Coordination Game?*

In Chapter 5 we discussed the impact of introducing bias in the choice of agents in each round. For example, when between a *blue* and a *red* node, an agent might choose *blue* with a probability higher than $\frac{1}{2}$ in case there is a positive bias towards opinion *blue*. Although there is an indication that this problem is hard for a general undirected graph [37], we provide solutions for cycles (Theorem 5.3.22). We show that such games have a correspondence with processes of annihilating random walks on graphs, which is sustained by the fact that the formula found in Theorem 5.3.22 is similar to the solution of a biased random walk on a path, also known as the gambler's ruin problem (Example 2.4.11).

A8 *Can every state in a Flag Coordination Game be reached from any other state with positive probability?*

The correspondence between consensus games on cycles and a process involving random-walking particles also allowed us to solve, for particular Flag Coordination Games, a known hard problem in general [1, 58, 8] : given two possible configurations, A and B, in a distributed system that changes its state at successive time steps, can a system that started in configuration A ever reach configuration B?

We showed that, for biased (or unbiased) Flag Coordination Games on cycles, the problem of reachability can be solved, i.e., by looking at the equivalent process involving random walks, we can establish whether a given state is reachable from another. In sum, because random-walking particles annihilate each other as they meet, and no new walk is created, configuration with more particles might reach one with less, but not the other way round.

Refer to Table 6.2 for a remainder of symbols and definitions for each set of rules of Flag Coordination Games. Finally, we present diagram in Figure 6.1 that summarises the relationship between the different set of rules of Flag Coordination Games, and highlight some of the theorems proven in this dissertation. Note that results in the diagram are contributions of this dissertation. The exceptions are Theorem 2.3.1 by Hassin and Peleg [35] and Theorem 2.4.17 by Cooper and Rivera [18], which were added to show how they relate with the models we studied. We also add some examples from Chapter 2, highlighting the fact that, although they can be seen as Flag Coordination Games, they are not seen as generalised consensus games.

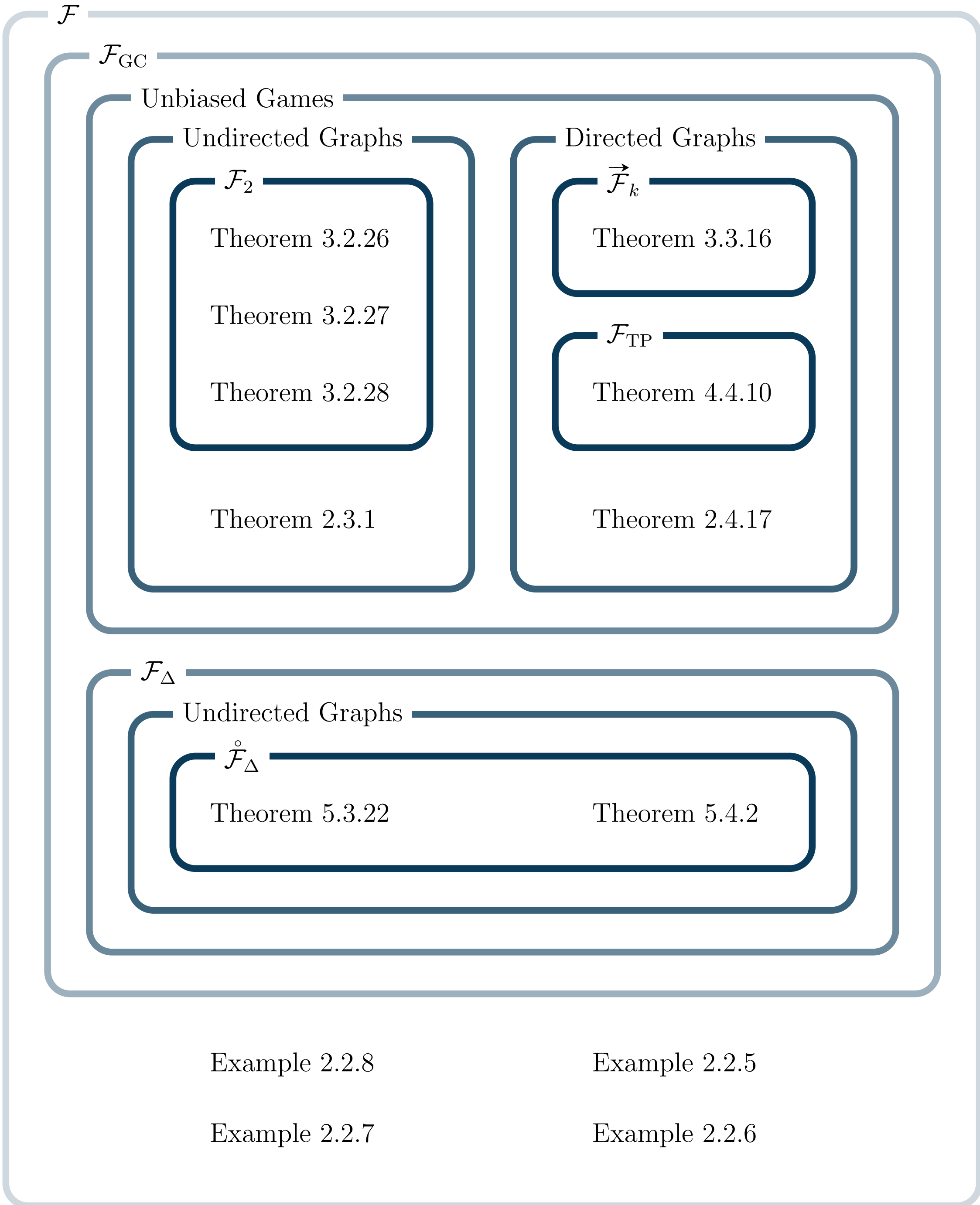


Figure 6.1: Diagram Depicting the Connection Between the Different Set of Rules for Flag Coordination Games Discussed in This Dissertation, Alongside Some Key Results.

$\mathcal{F}$	Flag Coordination Game	Definition 2.2.1
$\mathcal{F}_{\mathrm{GC}}$	Generalised Consensus Game	Definition 3.2.1
$\mathcal{F}_2$	Game on (Undirected) Bipartite Graphs	Definition 3.2.3
$\vec{\mathcal{F}}$	Generalised Consensus in Directed Graphs	Definition 3.3.2
$\vec{\mathcal{F}}_k$	Generalised Consensus in k-partite Digraphs	Definition 3.3.6
$\mathcal{F}_{\mathrm{TP}}$	Team Persuasion Game	Definition 4.3.2
$\mathcal{F}_{\Delta}$	Biased Generalised Consensus Game	Definition 5.3.1
$\mathring{\mathcal{F}}_{\Delta}$	Biased Two-colour Consensus Game on Cycles	Definition 5.3.4

Table 6.2: Notation of Each Set of Rules of Flag Coordination Games and Their Original Definitions.

6.2 Future Work

Even considering the several examples of Flag Coordination Games presented in Chapter 2, the wide range of possible applications does not seem to have yet been fully explored. Indeed, in this section we present a few more avenues of research bringing concepts and results of Flag Coordination Games into new areas. We also propose directions to further develop the results presented in Section 6.1.

6.2.1 Improvements on Results

In this section we study possible ramifications of our results in Chapters 3, 4, and 5.

Regarding results from Chapter 3, one avenue of future work is to find bounds for the expected duration of games in digraphs in a similar way to how we found bounds for undirected bipartite graphs. I would also be interesting to understand the effect of agents having longer memories in specific classes of graphs, such as cycles or complete graphs. Note that a general solution for the longer memories problem has been added to Appendix A. In that scenario, one could explore mixed networks, in which different agents are being able to remember different sets of previous rounds, and establish whether probabilities for reaching consensus increase or decrease for each given memory profile. A more theoretical question regarding games on bipartite graphs the one regarding conditional expectations on the time for the process to finish. For example, given that a game will end successfully, what is its the expected time?

A interesting line of future work for not only generalised consensus games but Flag Coordination Games in general, is the consideration of malicious agents that seek to thwart the process to reach one of the goal states. It would be then necessary to allow other agents to have a longer memory to be able to try to infer (with some probability) whether a given agent is behaving as it should be.

Future work will generalise the techniques of Chapter 3 to the anti-consensus problem on a directed graph investigated in Chapter 4. Specifically, if the Team Persuasion game will reach a goal state, we can calculate the expected number of rounds until that happens. We could also investigate different generalisations of the team persuasion game. There are various assumptions on the digraph that we could modify. For example, generalising from bipartite to multipartite argumentation frameworks in which many teams seek to persuade the audience. Additionally, we can lift the assumption that no agent attacks its fellow agents of the same team. Such a team seems quite unlikely (and thus is not considered here), but occasionally this may occur, e.g. a campaigner who wishes to leave the EU because their environmental laws are too restrictive on UK businesses, and a campaigner who wishes to leave the EU because they do not have strong enough environmental laws; both campaigners would be on the same team, but their arguments are seemingly conflicting. Further generalisations include consideration of the case in which each agent can assert more than one argument or the consideration of heterogeneous agents in the same framework that can also be altruistic or timid. Ultimately, we hope such generalisations can give insight into situations in which individual goals and societal goals conflict to a greater extent, and how this conflict can be resolved.

In future work, the results for team persuasion games can be applied to other types of argument dialogue games, such as negotiations [63]. Although team persuasion games are similar to real-world political debates in which bribery is common, there are other forms of dialogue in which it might also occur, such as deliberations.

Considering games which do not become state-stable, it would be interesting to investigate (1) in what proportion of rounds is the topic acceptable, and (2) what is the probability that the topic is acceptable at a specific round in the future. With respect to the first question, we might determine the winning team to be the one who makes the topic acceptable/unacceptable in the majority of rounds. The second question is particularly interesting in the context of referendum-like domains, in which there is a set date (round t) in which the audience determines whether the topic is acceptable (and thus which team wins): in this case it does not matter whether there is state-stability, only that the topic is acceptable in round t.

We could also extend the results for other types of team persuasion game, such as by generalising the result of Proposition 4.5.9 to an n-AF with an arbitrary number of PSNE. Currently, our model of bribery games assumes that bribers have no choice but to play; given that the briber's payoffs might be negative, it might be interesting to consider giving the bribers the option not to play the game, by including this in the game-theoretic analysis. Finally, by introducing a measure on the set of rounds in which the topic is accepted, we could study the long-run density of a given result in a distributed argumentation process.

With regards to biased consensus games, one could explore different graph structures for which biased consensus games posses analytic solutions regarding the probabilities involved in the process, as well as the complexity of the absorbing time. An ideal next candidate are path graphs, given its similarities with cycles in particular with regards to the correspondence to processes involving random-walking particles. Other good candidates are tree and star graphs. Another improvement on results from Chapter 5 is to find time bounds for biased consensus games on cycles to finish as well as extending the analysis to allow more than two colours.

In relation to the multiple consecutive games problem proposed in Section 5.4.3, we might be able to use tools such as drift on random walks to try to approach questions regarding probabilities of absorption or time bounds. This drift will take into account the position of the random walk, not the time in which it acts. Reachability is studied by Dunne and Chevaleyre [27] in the context of distributed negotiation schemes, in which we are interested in whether a desired allocation of resources can be reached from an initial one by local reallocations. It might be possible to related such processes with the reachability problem on cycles discussed in Section 5.4.1. Still on the reachability problem involving random-walking particles, a possible refinement of our work is to combine results by Grigoriev and Priezzhev in [33] with Theorem 5.4.2 to obtain not only whether a state A is reachable by state B, but what is the probability of that occurring.

6.2.2 Future Applications of Flag Coordination Games

The identification of collusion in competitive markets is a known problem for state agencies. Indeed, most western countries outlaw collusion between competing companies and regulate this activity through agencies such as the Competition and Markets Authorities in the UK. Seeking to broaden the spectrum of possible applications of Flag Coordination Games, we might be able to formalise this process using a sort of anti-consensus algorithm. Nodes in a complete graph would represent market agents that have to avoid collusion as well as avoiding appearing like they are colluding, regardless of their intentions. For that, they can follow an algorithm that just avoids them sharing the same state (representing a monetary value) of any other agent. There would be restrictions on which value to change to. The following example provides a formal definition of a Flag Coordination Game that could be used to model collusion.

EXAMPLE 6.2.1 (COLOURING OF COMPLETE GRAPHS). Consider a Flag Coordination Game $\mathcal{F} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A} \rangle$ played in a complete graph $G = K_n$. Nodes aim to proper colour this graph in a synchronous way (and discrete time-set T). Clearly, at least n colours are needed, but not necessarily available to all nodes during all rounds. Indeed, we assume that the colours available to a node in a given round depends on its current state. The way we depict this relation is by a digraph $H = (V(H), E(H))$ in which $V(H) = X$ is the set of colours in this Flag Coordination Game and a direct edge $(x_i, x_j) \in E(H)$ represents that if agent $v \in V(G)$ is currently coloured x_i, i.e., $S_t(v) = x_i$, then v can choose colour x_j for the next round. Formally, v chooses randomly from the set $\{x \mid (S_t(v), x) \in E(H)\}$. The algorithm of a node v is to randomly select one of the colours currently at its disposal if some neighbour (all other nodes in K_n are neighbours of v) is currently coloured the same as v. Otherwise, v keeps its colour.

In these conditions, we are interested in how long the process is going to take as a function of the number of nodes in V, and also in which colours end up being used at the end, i.e., how far agents' states *travelled* in graph H.

Considering H as a very long (or infinite) path with nodes representing monetary values with as many decimal places wanted, we can devise a criterion to check whether market agents are indeed avoiding collusion. For that, we apply the work by Cooper *et al.* [17], in which dispersion processes are studied, including time bounds and expected width of the dispersion in different graph structures. More specifically, they study processes that start with a given number of random-walking particles all on a given node on a graph, say, a path. At each step, if the particles are not alone, they move to a random neighbour (regardless of this neighbouring node being empty or not). The process ends whenever all particles are alone. Under these conditions, if there are n initial random walks, dispersion will take $O(n^3 \log n)$ and the distance from the origin D_{disp} will be, with high probability, for any $\epsilon > 0$, such that $\lfloor \frac{n}{2} \rfloor \leq D_{\text{disp}} \leq 4(1+\epsilon) n \log n$ [17, Theorem 4].

Appendix A

Agents with Longer Memories

In this section, we provide a solution for the scenario in which agents can remember past rounds and may take in account previous colours of some of the other nodes. More formally, we present the following more general version of a consensus game:

Definition A.0.1 *Let $\overrightarrow{\mathcal{F}_\psi} = \langle G, X, T, \Gamma, \phi, \beta, \psi, \sigma, \mathcal{A}\rangle$ be a generalised consensus game as in Definition 3.3.2 with the difference that we allow agents to have longer memories. Although, different nodes might have different memory depths, we assume that a given node has constant memory throughout a game.*

In the following theorem, we are going to show that it is possible to model a longer memory process as a memory-less one played in a suitable graph G'. In loose terms, we are going to create G' by combining m copies of G, where m is the maximum depth of memory among all the nodes in G. These copies are numbered $0, -1, \dots, -m$, to indicate that layer $k < 0$ records the configuration of the game on G k rounds prior to the current one. For that, we need to have nodes i in layer $k \neq 0$, denoted by v_i^k, to have an unique edge pointing from it to its copy in layer $k+1$. The out-matrix of G' will have the following structure:

$$H' = \left(\begin{array}{c:c:c:c} H_0 & H_{-1} & \dots & H_{-m} \\ \hdashline I & 0 & 0 & 0 \\ \hdashline 0 & \ddots & \ddots & 0 \\ \hdashline 0 & 0 & I & 0 \end{array}\right) \tag{A.1}$$

Theorem A.0.2 *Let $(\overrightarrow{\mathcal{F}_\psi}, S_0)$ be a game as in Definition A.0.1, and let $m = \max_{v\in V}\{\psi(v)\}$. Consider the graph G' as defined above with layers $G_0, G_{-1}, \dots, G{-}m$.*

Then, the probability that consensus is achieved in G is given by the probability that consensus is achieved in G', where the configuration of G' is either given by the previous rounds of the game or, if not enough rounds were played, by the expected colours of each node as given by Chapman–Kolmogorov equations applied to the initial configurations.

Proof. There are two cases to be considered. The first assumes we have the record of at least m rounds of this game. In this case, it is enough to evaluate the stationary distribution of H' and apply theorem 3.3.16. Note that for consensus to be achieved in G', we need G_0 to have reached consensus, and thus although there might some time delay between G_0 reaching consensus and all the other layers of G' capturing this consensus, we know that consensus will be reached in G if and only if it is reached in G'.

The second case is when we do not have enough previous rounds to construct the initial configuration of G'. In that case, we take the initial configuration of G and copy it to G_{-m}. Then, subsequently apply the Chapman-Kolmogorov equations to the bottom layers to get the expected probability that nodes in layer k will have the colour x. ■

EXAMPLE A.0.3. Consider a consensus game played on $G := G_0 = C_4$ in which nodes have probability p of copying the current colour of one of its neighbours (uniformly at random), and probability $(1-p)$ of copying the previous round's colour of of one of its neighbours (uniformly at random). In this case, we have

$$\psi(v,t) = 1 \text{ for all } v \in V \text{ and } t > 0$$

We now have to find the stationary distribution of the out-matrix of the new graph G', where $G'(V) = G_0(V) \cup G_{-1}(V)$, $G_{-1} = C_4$, and edges of G' are given by the out-matrix

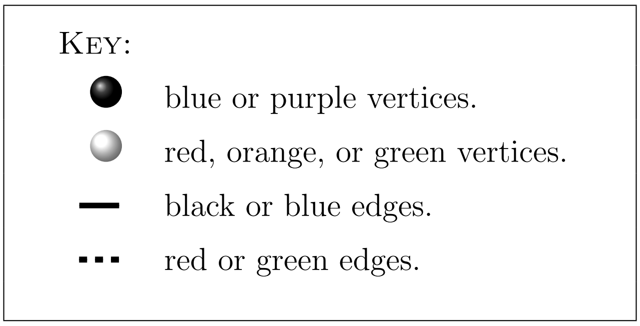
$$H = \begin{bmatrix} 0 & \frac{p}{2} & 0 & \frac{p}{2} & 0 & \frac{1-p}{2} & 0 & \frac{1-p}{2} \\ 0 & \frac{p}{2} & 0 & \frac{p}{2} & 0 & \frac{1-p}{2} & 0 & \frac{1-p}{2} \\ \frac{p}{2} & 0 & \frac{p}{2} & 0 & \frac{1-p}{2} & 0 & \frac{1-p}{2} & 0 \\ \frac{p}{2} & 0 & \frac{p}{2} & 0 & \frac{1-p}{2} & 0 & \frac{1-p}{2} & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{A.2}$$

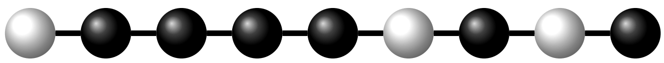
The (normalised) stationary distribution of H is given by

$$\mu = \frac{1-p}{4p}\left(\frac{1}{1-p}, \frac{1}{1-p}, \frac{1}{1-p}, \frac{1}{1-p}, 1, 1, 1, 1\right) \tag{A.3}$$

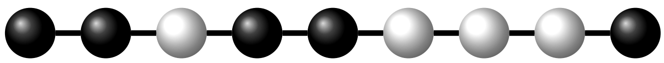
Appendix B

Black and White Figures





(a) Configuration A.



(b) Configuration B.

Figure B.1: Two possible configurations of Robot Bucket Brigade. (**black and white version of Figure 2.1**)

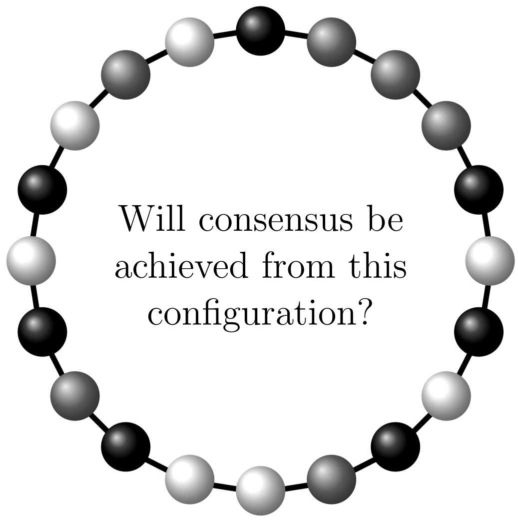


Figure B.2: Consensus Game on a Cycle C_{20} with 3 Colours.
(black and white version of Figure 3.1)

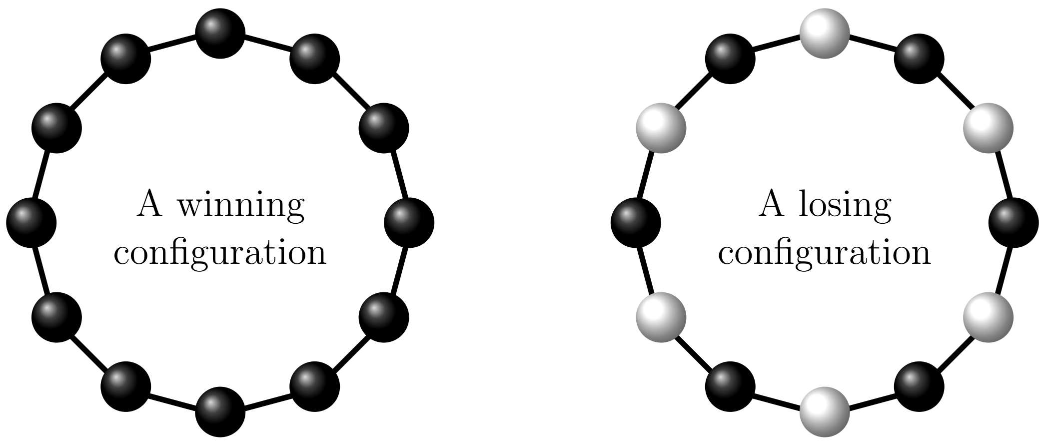


Figure B.3: A Consensus in Blue (left) and a Configuration from which Consensus Will Never be Achieved (right) on a Cycle C_{12}.
(black and white version of Figure 3.2)

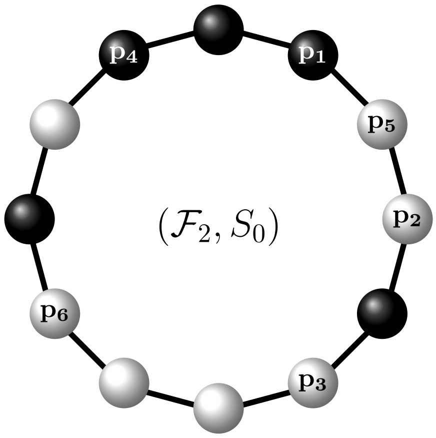


Figure B.4: Initial states of a Flag Coordination Game, and its Correspondent Annihilating Random-Walking Particles, Depicted in the Same Graph. Nodes with p_i Indicate the Presence of Random Walking Particle i on that Node.
(**black and white version of Figure 3.3**)

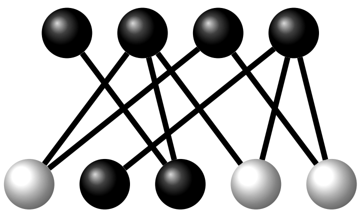


Figure B.5: Example of a Single-partition Round.
(**black and white version of Figure 3.4**)

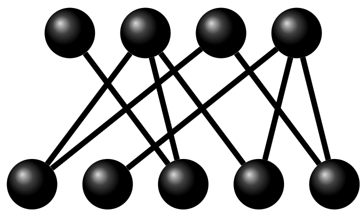


Figure B.6: Only Reachable Consensus From Game in Figure 3.4.
(**black and white version of Figure 3.5**)

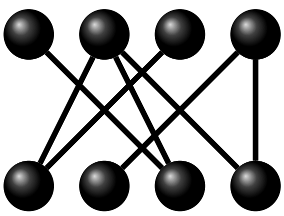


Figure B.7: Example of a Winning Configuration.
(**black and white version of Figure 3.6**)

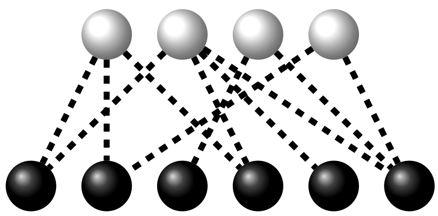


Figure B.8: Example of a Losing Configuration.
(**black and white version of Figure 3.7**)

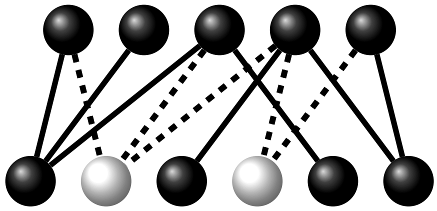


Figure B.9: Game $(\mathcal{F}_2, S_0)$ as in Example 3.2.23.
(black and white version of Figure 3.8)

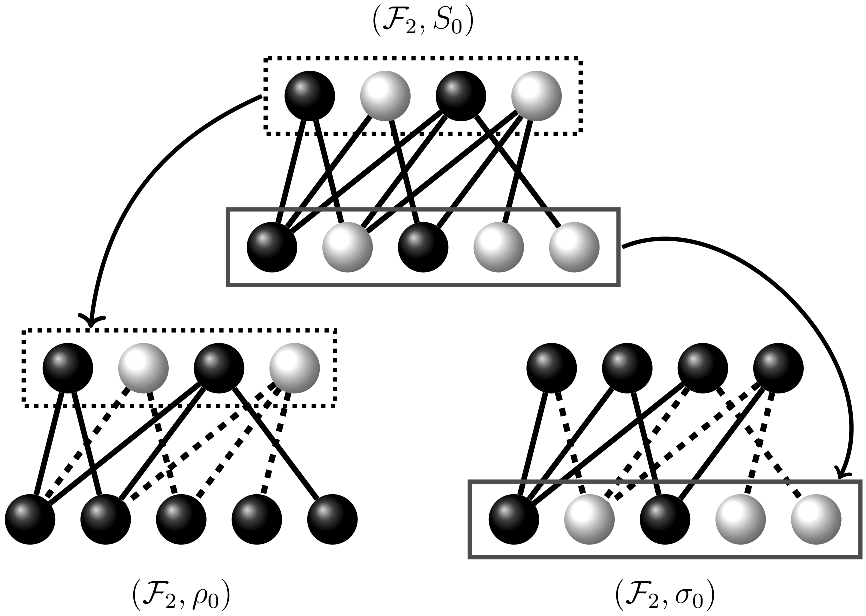


Figure B.10: Example of Game $(\mathcal{F}_2, S_0)$ Being Split in $(\mathcal{F}_2, \sigma_0)$ and $(\mathcal{F}_2, \rho_0)$.
(black and white version of Figure 3.9)

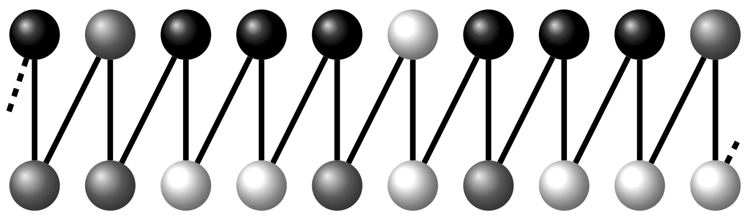
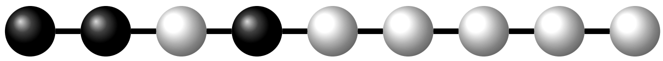
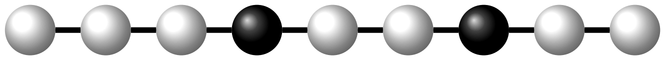


Figure B.11: Alternative Display of the Cycle in Figure 3.1 Evidencing Partitions of G.
(black and white version of Figure 3.10)



(a) Configuration A'.



(b) Configuration B'.

Figure B.12: Translation of Robot Bucket Brigades Configurations Into Consensus Games.

(black and white version of Figure 3.11)

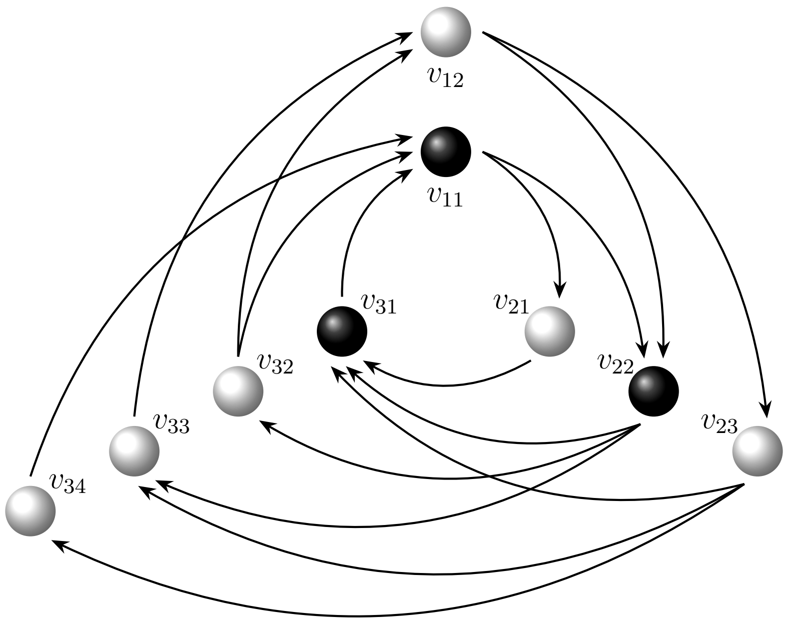


Figure B.13: A Generalised Consensus Game $(\vec{\mathcal{F}}, S_0)$ in a Digraph G that Might Not Lead to Consensus.

(black and white version of Figure 3.12)

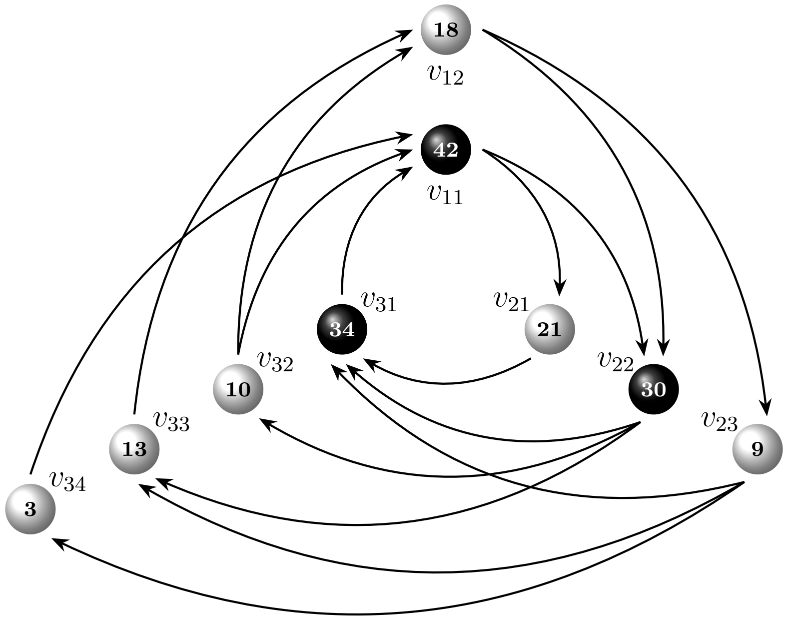


Figure B.14: Game $(\vec{\mathcal{F}}_3, S_0)$ with Influences of Each Node (Multiplied by 60 for Readability).

(black and white version of Figure 3.13)

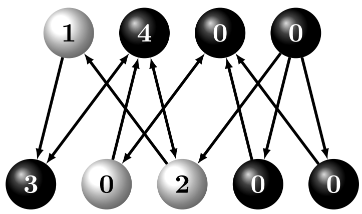


Figure B.15: A Game $(\vec{\mathcal{F}}, S_0)$ on a Weakly Connected Graph.

(black and white version of Figure 3.14)

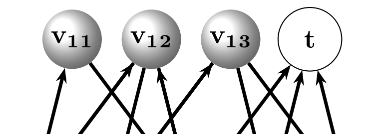


Figure B.16: The defenders' goal state γ_{for}; all defenders are asserting their argument.

(black and white version of Figure 4.2)

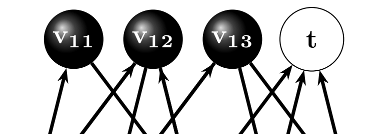


Figure B.17: The attackers' goal state γ_{ag}; all attackers are asserting their argument.

(black and white version of Figure 4.3)

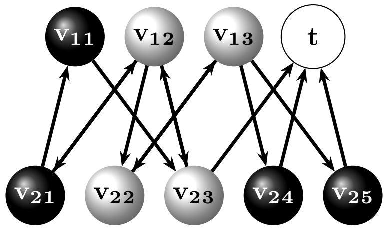


Figure B.18: An Initial Configuration $(\mathcal{F}_{\mathrm{TP}}, S_0)$ for the example in Figure 4.1.
(black and white version of Figure 4.4)

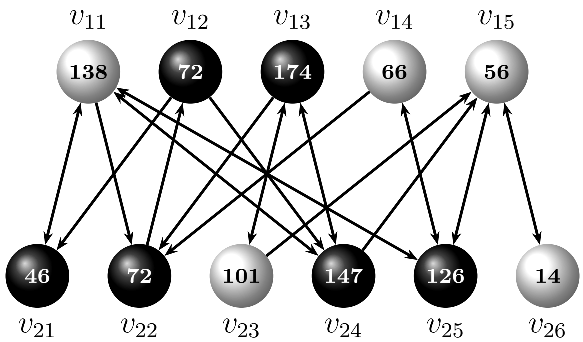


Figure B.19: The AF Underlying our Example. Current Colouring and Influences Depicted in Each Argument. Influences were Multiplied by 506 for readability.
(black and white version of Figure 4.6)

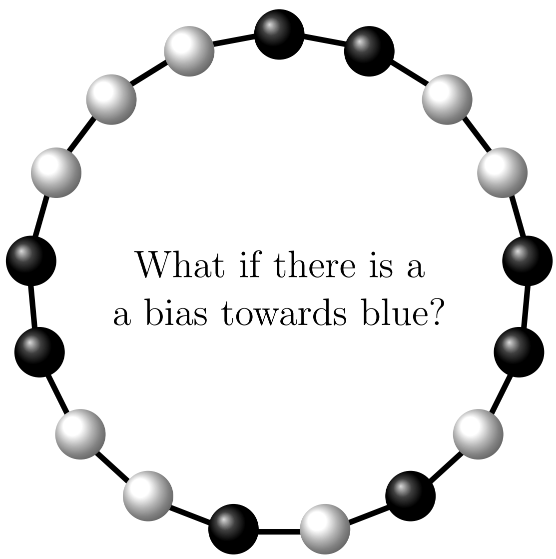


Figure B.20: Biased Consensus Game on a 17-Cycle.
(black and white version of Figure 5.1)

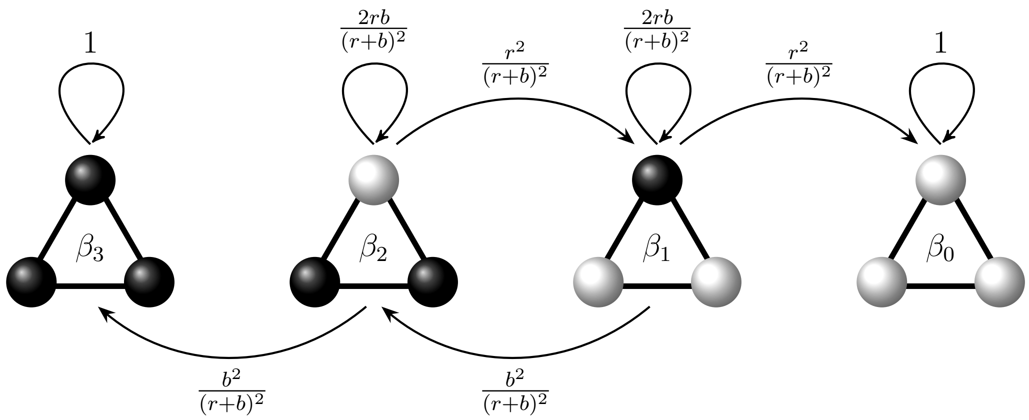


Figure B.21: Possible States and Their Transition Probabilities of a Biased Consensus Game on C_3.
(black and white version of Figure 5.2)

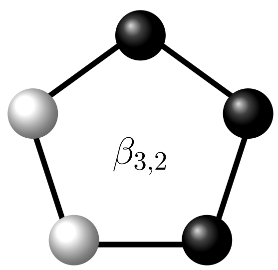


Figure B.22: Configuration $\beta_{3,2}$ of C_5.
(black and white version of Figure 5.3)

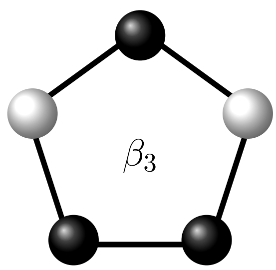


Figure B.23: Configuration β_3 of C_5.
(black and white version of Figure 5.4)

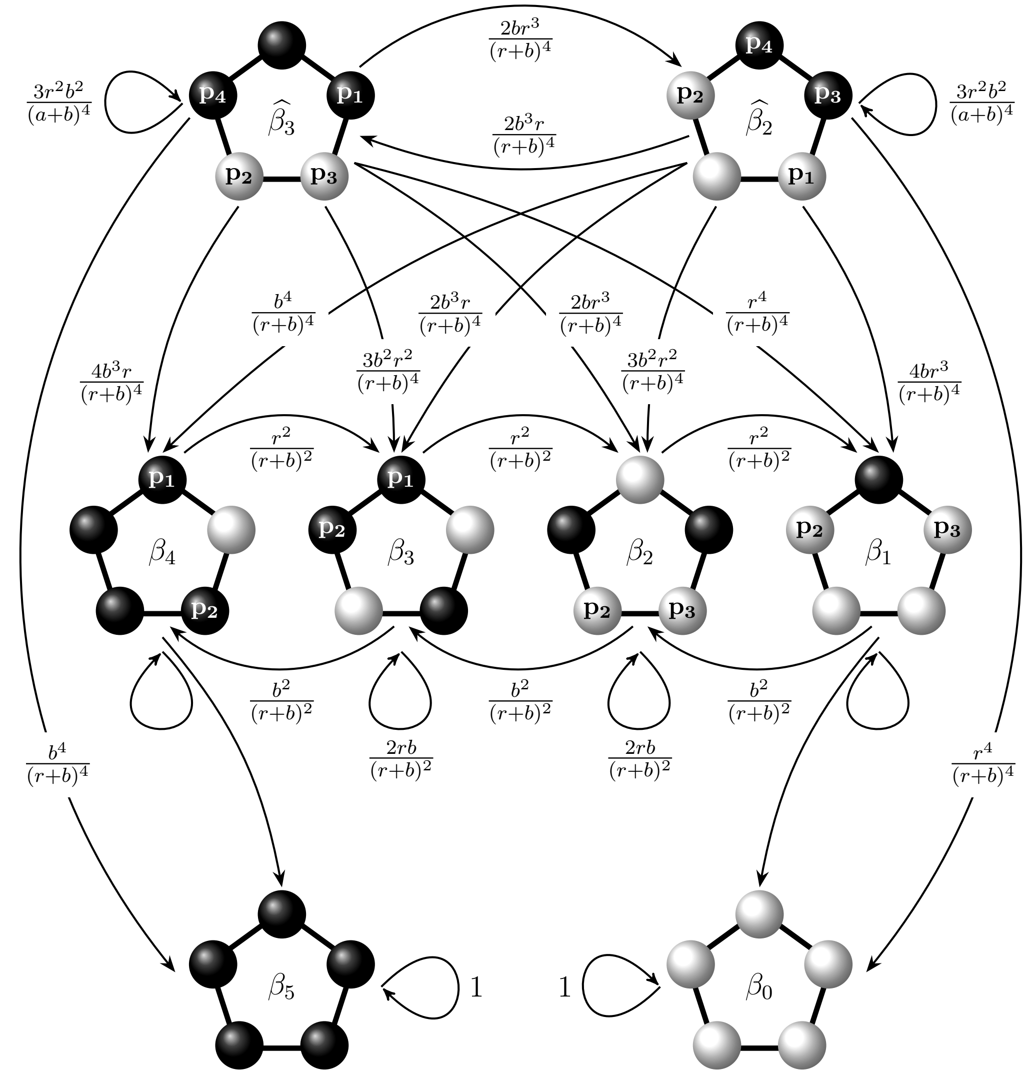


Figure B.24: Possible States Up To Symmetry and Their Transition Probabilities of a Biased Consensus Game on C_5.

(black and white version of Figure 5.5)

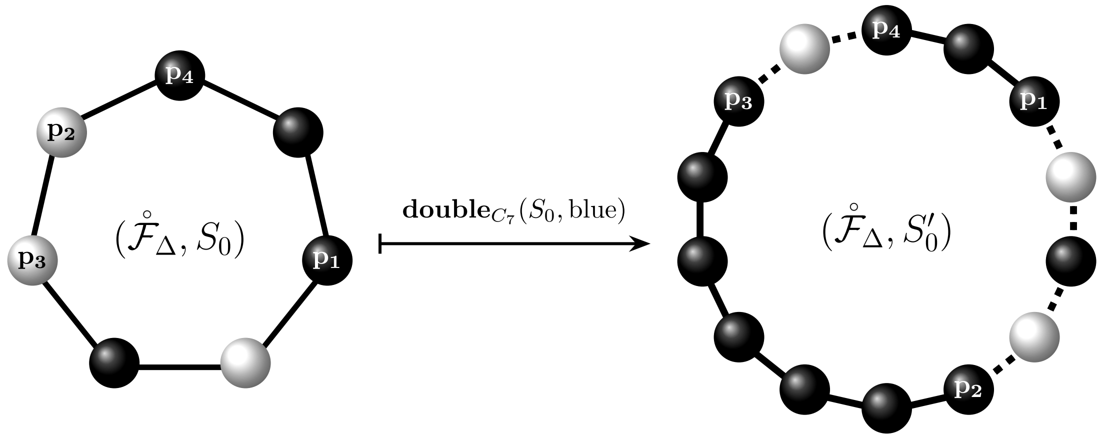


Figure B.25: Application of $\mathbf{double}_{C_7}(S_0, \text{blue})$. Randomising nodes have particles on them, represented by p_1, p_2, p_3, and p_4.

(black and white version of Figure 5.6)

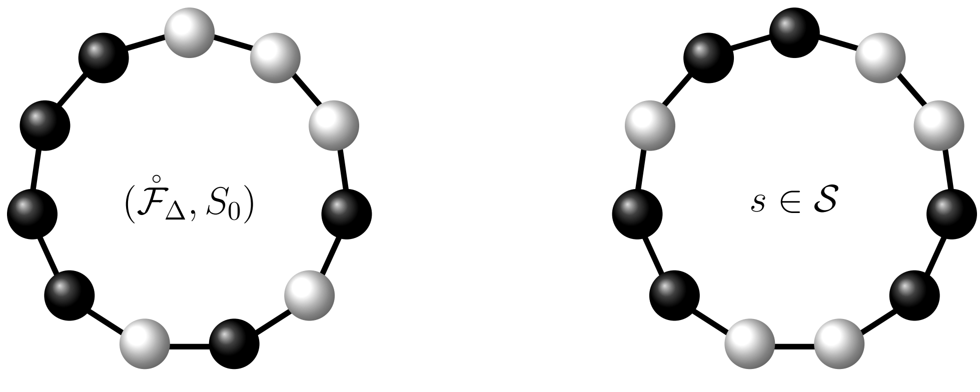


Figure B.26: The Initial Configuration of a Game $(\mathring{\mathcal{F}}_\Delta, S_0)$ and a Given Configuration $s \in \mathcal{S}$. Can $(\mathring{\mathcal{F}}_\Delta, S_0)$ Eventually Reach s?

(black and white version of Figure 5.7)

Bibliography

[1] S Akshay, Timos Antonopoulos, Joël Ouaknine, and James Worrell. Reachability problems for Markov chains. *Information Processing Letters*, 115(2):155–158, 2015.

[2] Krzysztof R Apt and Ehsan Shoja. Self-stabilization through the lens of game theory. In *It's All About Coordination*, pages 21–37. Springer, 2018.

[3] John Baez and James Dolan. Categorification. *Higher Category Theory*, pages 1–36, 1998.

[4] Jon Barwise. Scenes and other situations. *Journal of Philosophy*, 78(7):369–397, 1981.

[5] Trevor Bench-Capon. Persuasion in practical argument using value-based argumentation frameworks. *Journal of Logic and Computation*, 13(3):429–448, 2003.

[6] Dimitri P Bertsekas and John N Tsitsiklis. *Parallel and Distributed Computation: Numerical Methods*, volume 23. Prentice Hall Englewood Cliffs, NJ, 1989.

[7] Vincent D Blondel, Julien M Hendrickx, Alex Olshevsky, and John N Tsitsiklis. Convergence in multiagent coordination, consensus, and flocking. In *Decision and Control, 2005 and 2005 European Control Conference. CDC-ECC'05. 44th IEEE Conference on*, pages 2996–3000. IEEE, 2005.

[8] Vincent D Blondel and Natacha Portier. The presence of a zero in an integer linear recurrent sequence is NP-hard to decide. *Linear Algebra and its Applications*, 351:91–98, 2002.

[9] Gustavo Bodanza, Fernando Tohmé, and Marcelo Auday. Collective argumentation: A survey of aggregation issues around argumentation frameworks. *Argument & Computation*, 8(1):1–34, 2017.

[10] Elise Bonzon and Nicolas Maudet. On the outcomes of multiparty persuasion. In *Proceedings of the 8th International Workshop on Argumentation in Multi-Agent Systems*, pages 86–101. Springer, 2012.

[11] Bojana Borovicanin, Kinkar Ch Das, Boris Furtula, and Ivan Gutman. Bounds for Zagreb Indices. *MATCH Communications in Mathematical and in Computer Chemistry*, 78(1):17–100, 2017.

[12] Pierre Brémaud. *Discrete Probability Models and Methods*, volume 78. Springer, 2017.

[13] Martin Caminada. On the issue of reinstatement in argumentation. *Logics in Artificial Intelligence*, 4160:111–123, 2006.

[14] Kamalika Chaudhuri, Fan Chung Graham, and Mohammad Shoaib Jamall. A network coloring game. In *International Workshop on Internet and Network Economics*, pages 522–530. Springer, 2008.

[15] Colin Cooper, Robert Elsasser, Hirotaka Ono, and Tomasz Radzik. Coalescing random walks and voting on connected graphs. *SIAM Journal on Discrete Mathematics*, 27(4):1748–1758, 2013.

[16] Colin Cooper, Robert Elsässer, and Tomasz Radzik. The power of two choices in distributed voting. In *41st International Colloquium on Automata, Languages, and Programming*, pages 435–446. Springer, 2014.

[17] Colin Cooper, Andrew McDowell, Tomasz Radzik, Nicolás Rivera, and Takeharu Shiraga. Dispersion processes. *Random Structures & Algorithms*, 53(4):561–585, 2018.

[18] Colin Cooper and Nicolas Rivera. The linear voting model. In *43rd International Colloquium on Automata, Languages, and Programming (ICALP 2016)*. Schloss Dagstuhl-Leibniz-Zentrum fuer Informatik, 2016.

[19] Kinkar Ch Das. Maximizing the sum of the squares of the degrees of a graph. *Discrete Mathematics*, 285(1-3):57–66, 2004.

[20] Kinkar Ch Das, Kexiang Xu, and Junki Nam. Zagreb indices of graphs. *Frontiers of Mathematics in China*, 10(3):567–582, 2015.

[21] Dominique de Caen. An upper bound on the sum of squares of degrees in a graph. *Discrete Mathematics*, 185(1-3):245–248, 1998.

[22] Morris H DeGroot. Reaching a consensus. *Journal of the American Statistical Association*, 69(345):118–121, 1974.

[23] Frank P. M. Dignum and Gerard A. W. Vreeswijk. Towards a testbed for multi-party dialogues. *Advances in Agent Communication*, pages 212–230, 2004.

[24] Edsger W Dijkstra. Self-stabilizing systems in spite of distributed control. *Communications*, 1974.

[25] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial Intelligence*, 77:321–357, 1995.

[26] Paul E Dunne. Computational properties of argument systems satisfying graph-theoretic constraints. *Artificial Intelligence*, 171(10-15):701–729, 2007.

[27] Paul E Dunne and Yann Chevaleyre. The complexity of deciding reachability properties of distributed negotiation schemes. *Theoretical Computer Science*, 396(1-3):113–144, 2008.

[28] Paul E Dunne, Anthony Hunter, Peter McBurney, Simon Parsons, and Michael Wooldridge. Weighted argument systems: Basic definitions, algorithms, and complexity results. *Artificial Intelligence*, 175(2):457, 2011.

[29] Rick Durrett. *Probability: Theory and Examples.* Cambridge University Press, 2010.

[30] Martin Gardner. Mathematical games. *Scientific American*, 222(6):132–140, 1970.

[31] Jacob Glazer and Ariel Rubinstein. Debates and decisions: On a rationale of argumentation rules. *Games and Economic Behavior*, 36(2):158–173, 2001.

[32] Jacob Glazer and Ariel Rubinstein. On optimal rules of persuasion. *Econometrica*, 72(6):1715–1736, 2004.

[33] S Yu Grigoriev and Vyacheslav Borisovich Priezzhev. Random walk of annihilating particles on the ring. *Theoretical and Mathematical Physics*, 146(3):411–420, 2006.

[34] Ivan Gutman and Nenad Trinajstić. Graph theory and molecular orbitals. total φ-electron energy of alternant hydrocarbons. *Chemical Physics Letters*, 17(4):535–538, 1972.

[35] Yehuda Hassin and David Peleg. Distributed probabilistic polling and applications to proportionate agreement. *Information and Computation*, 171(2):248–268, 2001.

[36] Anthony Hunter. Toward higher impact argumentation. In *Proceedings of the 19th American National Conference on Artificial Intelligence*, pages 275–280. MIT Press, 2004.

[37] Rasmus Ibsen-Jensen, Krishnendu Chatterjee, and Martin A Nowak. Computational complexity of ecological and evolutionary spatial dynamics. *Proceedings of the National Academy of Sciences*, 112(51):15636–15641, 2015.

[38] Ernst Ising. Beitrag zur theorie des ferromagnetismus. *Zeitschrift für Physik*, 31(1):253–258, 1925.

[39] Matthew O Jackson. *Social and economic networks*. Princeton university press, 2010.

[40] Magdalena Kacprzak, Marcin Dziubinski, and Katarzyna Budzynska. Strategies in Dialogues: A Game-Theoretic Approach. In *Computational Models of Argument (COMMA)*, pages 333–344. IOS Press, 2014.

[41] Michael Kearns, Stephen Judd, Jinsong Tan, and Jennifer Wortman. Behavioral experiments on biased voting in networks. *Proceedings of the National Academy of Sciences*, 106(5):1347–1352, 2009.

[42] Michael Kearns, Siddharth Suri, and Nick Montfort. An experimental study of the coloring problem on human subject networks. *Science*, 313(5788):824–827, 2006.

[43] David Kohan Marzagão, Josh Murphy, Anthony Peter Young, Marcelo M Gauy, Michael Luck, Peter McBurney, and Elizabeth Black. Team Persuasion. In *The*

3rd International Workshop on Theory and Applications of Formal Argument (TAFA), pages 159–174. Springer, 2017.

[44] David Kohan Marzagão, Nicolás Rivera, Colin Cooper, Peter McBurney, and Kathleen Steinhöfel. Multi-agent flag coordination games. In *Proceedings of the 16th Conference on Autonomous Agents and MultiAgent Systems (AAMAS)*, pages 1442–1450. International Foundation for Autonomous Agents and Multiagent Systems, 2017.

[45] Paul Krause, Simon Ambler, Morten Elvang-Goransson, and John Fox. A logic of argumentation for reasoning under uncertainty. *Computational Intelligence*, 11(1):113–131, 1995.

[46] Fabian Kuhn and Rogert Wattenhofer. On the complexity of distributed graph coloring. In *Proceedings of 25th ACM Symposium on Principles of Distributed Computing*, pages 7–15. ACM, 2006.

[47] Leslie Lamport. Solved problems, unsolved problems and non-problems in concurrency. *ACM SIGOPS Operating Systems Review*, 19(4):34–44, 1985.

[48] Erez Lieberman, Christoph Hauert, and Martin A Nowak. Evolutionary dynamics on graphs. *Nature*, 433(7023):312, 2005.

[49] Paul-Amaury Matt and Francesca Toni. A game-theoretic measure of argument strength for abstract argumentation. In *European Workshop on Logics in Artificial Intelligence*, pages 285–297. Springer, Springer, 2008.

[50] Barry M McCoy and Tai Tsun Wu. *The two-dimensional Ising model*. Courier Corporation, 2014.

[51] Mihail Mihaylov. *Decentralized Coordination in Multi-Agent Systems*. PhD thesis, Vrije Universiteit Brussel, 2012.

[52] Mihail Mihaylov, Karl Tuyls, and Ann Nowé. A decentralized approach for convention emergence in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 28(5):749–778, 2014.

[53] Dov Monderer and Lloyd Shapley. Potential games. *Games and Economic Behavior*, 14(1):124–143, 1996.

[54] Patrick Alfred Pierce Moran. Random processes in genetics. In *Mathematical Proceedings of the Cambridge Philosophical Society*, pages 60–71. Cambridge University Press, 1958.

[55] John F. Nash. Equilibrium points in n-person games. *Proceedings of the National Academy of Sciences*, 36(1):48–49, 1950.

[56] John F. Nash. Non-cooperative games. *Annals of Mathematics*, pages 286–295, 1951.

[57] Reza Olfati-Saber, J. Alex Fax, and Richard M. Murray. Consensus and cooperation in networked multi-agent systems. *Proceedings of IEEE*, 95(1):215–233, 2007.

[58] Joël Ouaknine and James Worrell. Positivity problems for low-order linear recurrence sequences. In *Proceedings of the Twenty-Fifth Annual ACM-SIAM Symposium on Discrete Algorithms*, pages 366–379. Society for Industrial and Applied Mathematics, 2014.

[59] Panagiota N Panagopoulou and Paul G Spirakis. A game theoretic approach for efficient graph coloring. In *International Symposium on Algorithms and Computation*, pages 183–195. Springer, 2008.

[60] Henry Prakken. Formal systems for persuasion dialogue. *The Knowledge Engineering Review*, 21(02):163–188, 2006.

[61] Martin L Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2014.

[62] Iyad Rahwan and Kate Larson. Mechanism design for abstract argumentation. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 2*, pages 1031–1038. International Foundation for Autonomous Agents and Multiagent Systems, IFAAMS, 2008.

[63] Iyad Rahwan, Sarvapali D. Ramachurn, Nicholas R. Jennings, Peter McBurney, Simon Parsons, and Liz Sonenberg. Argumentation-based negotiation. *The Knowledge Engineering Review*, 18(4):343–375, 2003.

[64] Wei Ren, Randal W Beard, and Ella M Atkins. A survey of consensus problems in multi-agent coordination. In *Proceedings of 2005 American Control Conference, 2005*, pages 1859–1864. IEEE, 2005.

[65] José Carlos Rosales and Pedro A. García-Sánchez. *Numerical Semigroups*, volume 20. Springer Science & Business Media, 2009.

[66] Alain Sarlette. *Geometry and Symmetries in Coordination Control*. PhD thesis, Université de Liège, 2009.

[67] Alain Sarlette, Rodolphe Sepulchre, and Naomi Ehrich Leonard. Autonomous rigid body attitude synchronization. In *2007 46th IEEE Conference on Decision and Control*, pages 2566–2571. IEEE, 2007.

[68] Alain Sarlette, Sezai Emre Tuna, Vincent Blondel, and Rodolphe Sepulchre. Global synchronization on the circle. In *Proceedings of the 17th IFAC World Congress*, pages 9045–9050, 2008.

[69] Carl Shapiro and Hal Varian. *Information Rules: A Strategic Guide to the Network Economy*. Harvard Business Review Press, Cambridge, MA, USA, 1998.

[70] Hyun Song Shin. The burden of proof in a game of persuasion. *Journal of Economic Theory*, 64:253–264, 1994.

[71] Hyun Song Shin. Adversarial and inquisitorial procedures in arbitration. *RAND Journal of Economics*, 29:378–405, 1998.

[72] Peverill Squire. Why the 1936 literary digest poll failed. *Public Opinion Quarterly*, 52(1):125–133, 1988.

[73] Matthias Thimm. Strategic argumentation in multi-agent systems. *Künstliche Intelligenz*, 28:159–168, 2014.

[74] John Tsitsiklis, Dimitri Bertsekas, and Michael Athans. Distributed asynchronous deterministic and stochastic gradient optimization algorithms. *IEEE Transactions on Automatic Control*, 31(9):803–812, 1986.

[75] John N Tsitsiklis. *Problems in Decentralized Decision Making and Computation*. PhD thesis, Massachusetts Institute of Technology, 1984.

[76] Tamás Vicsek, András Czirók, Eshel Ben-Jacob, Inon Cohen, and Ofer Shochet. Novel type of phase transition in a system of self-driven particles. *Physical Review Letters*, 75(6):1226, 1995.

[77] Tim Weitzel. *Economics of Standards in Information Networks*. Physica, Heidelberg, Germany, 2004.

[78] Sheng-gui Zhang and Chun-cao Zhou. Bipartite graphs with the maximum sum of squares of degrees. *Acta Mathematicae Applicatae Sinica (English Series)*, 3:022, 2014.

[79] Bo Zhou. Remarks on Zagreb indices. *MATCH Communications in Mathematical and in Computer Chemistry*, 57:591–596, 2007.

List of Figures

Index

List of Symbols

$\mathcal{F}$	Flag Coordination Game
$\mathcal{F}_{\mathrm{GC}}$	Generalised Consensus Games
$\mathcal{F}_2$	Games on (Undirected) Bipartite Graphs
$\vec{\mathcal{F}}$	Generalised Consensus in Directed Graphs
$\vec{\mathcal{F}}_k$	Generalised Consensus in k-partite Digraphs
$\mathcal{F}_{\mathrm{TP}}$	Team Persuasion Game
$\mathcal{F}_{\Delta}$	Biased Generalised Consensus Game
$\mathring{\mathcal{F}}_{\Delta}$	Biased Two-colour Consensus Game on Cycles
S	State of a round of a game
T	Time-set
t	Time
X	Set of Colours
x	A colour of set X
$\mathcal{A}$	Set of algorithms
α	Algorithm
ϕ	Visibility of an agent
β	Flags of an agent
ψ	Memory of an agent
σ	Scheduler
Γ	Set of goal states
γ	Goal state
μ	Stationary distribution
Θ	Sum of influences
$\mathcal{K}$	Set of strongly connected components
Δ	Set of biases
δ	Bias
$\mathbb{N}$	Set of natural numbers
$\mathbb{Z}$	Set of integers
$\mathbb{R}$	Set of real numbers
$\mathbb{Q}$	Set of rational numbers
$\mathbb{P}$	Power set
$\mathcal{N}$	Neighbourhood of a vertex
AF	Argumentation framework
τ	Duration of a process