Imperial College London

Game Theoretic and Data-Driven Methods for Dynamic Decisions

- From Full to Partial Information -

Benita Alessandra Lucia Nortmann

January 2024

Department of Aeronautics

Imperial College London

Thesis submitted in part fulfilment of the requirements for the degree of Doctor of Philosophy of Imperial College London and the Diploma of Imperial College London

Abstract

Modern systems are increasingly complex, interconnected, and influenced by various decision makers. This introduces new challenges for designing dynamic decision laws which ensure such systems behave as expected, respond appropriately to inputs, or operate safely autonomously - particularly if decision makers have incomplete information regarding the system dynamics or performance criteria. The objective of this thesis is to develop game theoretic and data-driven methods for dynamic decisions towards tackling such challenges.

Dynamic game theory concerns the dynamic interaction of strategic decision makers called *players*. As games involve multi-objective optimisation problems, the “best” strategy for each player is typically not obvious, and various solution concepts exist. In this thesis, *feedback Nash equilibrium* solutions of linear quadratic discrete-time dynamic games are considered. Computing such solutions is generally challenging, and multiple solutions with different outcomes may exist. To build intuition, conditions characterising the number of solutions and certain properties are derived for games involving scalar dynamics. To address the challenges associated with obtaining solutions in the general case, a notion of approximate Nash equilibrium is introduced, and iterative Nash equilibrium finding methods are proposed.

Data-driven control exploits measured data to recover or replace missing information for designing dynamic decision laws. In this thesis, a framework to *design control laws directly using data*, while providing performance guarantees, is extended to the class of linear time-varying systems, and methods are developed to represent control objectives using data.

Combining the above, methods are proposed to overcome incomplete information in multi-player dynamic decisions. First, games in which one player lacks system and cost information are considered, before iterative data-driven methods are designed to determine a solution if all players have incomplete information.

The results are illustrated and motivated via numerical examples and practically relevant case studies, including macroeconomic policy design, power systems, snake-like robots, and human-robot interaction.

Statement of Originality

I confirm that this thesis contains my own original work, unless otherwise specified. To the best of my knowledge, all external sources and any work and ideas arising from collaboration with others have been appropriately acknowledged.

Copyright Statement

Acknowledgements

This thesis is the result of work carried out at the Department of Aeronautics at Imperial College London. Imperial Aeronautics is an amazing place full of incredibly smart and inspiring people and it has been a pleasure to be a part of it. I would like to thank the department for the great support and the opportunities they have provided over the past years. I would also like to thank EPSRC for funding my PhD research.

The journey through this doctoral degree has been a challenging yet rewarding experience. I would like to express my sincere gratitude to the people who have guided and accompanied me along the way. First and foremost, I thank my supervisor Dr Thulasi Mylvaganam for her guidance, her support, her patience and her encouragement and believe in me. It has been such a joy working with her and I have learnt so much. I would also like to thank Prof. Mario Sassano, Prof. Vijay Gupta, Prof. Claudio De Persis and Prof. Eric Kerrigan, as well as (former) fellow PhD students and collaborators Andrea Monti, Pranoy Das, Domenico Capello and Maria Scarpa for interesting discussions, insights, advice and recommendations.

Thank you to my examiners, Prof. Pietro Tesi and Prof. Thomas Parisini, for the discussion of the topic and for their feedback and recommendations for this thesis.

Thank you to the EEE lunch group, Alex, Omar, Angelo, Domenico, Federica and Mirko for welcoming me and for many lunches, coffees/hot waters, and climbing sessions. A massive thank you to all the amazing fellow Aero PhDs, in particular the basement crew Ali, Mario, Grace, Kazuki, James, Joanne and Kaloyan, as well as Lidia, Christian, Maria, Joséphine, Alexandra, Valentin, Francisco, Filippo, Massimiliano, Samet, Chi Hin, Henrik, Daniela and Stefanie for the lunches, work outs, tea breaks, mystery shots, trips and generally fun times making the last few years an unforgettable experience. Thank you to Imperial Shotokan Karate club, especially Sensei Sid and Zoltan, for providing a place to unwind, to focus and to have a fun time at training - I learned a lot from you.

A special thank you to Angelo, for bagging my keys, for having the same terrible humour as me, for many happy memories, for bearing with me when I have been working late, and for simply being a wonderful person all around, who I have been lucky to have around me throughout most of this time.

More generally, I would like to thank Abdul, for teaching me not to take life too seriously and for encouraging me to start this PhD journey. Thank you to Eugen and to Berk, for having stayed friends for all these years I have been away "vom See". Thank you to Rhys for always inspiring everyone around him to do their best, for his encouragement at the start of the PhD and for helping me to up my presentation skills. Even if contact has fizzled out a little, I would like to thank Naqiya for the Dosa lunches and deep Skype talks, and I would like to thank Laura and Annie, for having been good friends back at a time that really shaped me.

A massive thank you to my family, and particularly my mother, for always being there for me. You keep on supporting me and cheering for me through all my decisions - it means so much.

An extra shout out again to Ali for his motivational talks during the write-up, and to Angelo for proofreading parts of the thesis.

To my mother

‘You must do the thing you think you cannot do.’

Eleanor Roosevelt

Contents

List of Figures

List of Tables

List of Acronyms

LQ linear quadratic

LQR linear quadratic regulator

PDE partial differential equation

ODE ordinary differential equation

PID proportional-integral-derivative

LTI linear time-invariant

LMI linear matrix inequality

LTV linear time-varying

SDP semidefinite programme

NP nondeterministic polynomial-time

LAS locally asymptotically stable

LFT linear fractional transformation

DC direct current

AC alternating current

Chapter 1

Introduction

1.1 Motivation

From navigating social interactions to steering a car through traffic or managing inventory, dynamic decisions, big and small, are a constant part of our lives. Dynamic decision-making is characterised by taking place in real time in an environment which evolves over time, both due to past decisions and due to events and factors out of the influence of the decision maker. On a larger scale, situations involving dynamic decision-making include air traffic control, military strategy, economic policy making and managing emergency situations [1].

Deriving laws governing dynamic decisions is key to successful automation. Both in our everyday life and in the larger scale examples above, automation becomes ever more prevalent, such as the use of autonomous robots in disaster response [2]. The question is then how to derive decision laws to tackle complex challenges, such as to enable a robot to move autonomously through challenging environments or to interact safely with humans?

If the evolution of the decision-making process can be modelled as a dynamical system, *control theory*, the branch of engineering concerned with studying and shaping the behaviour of dynamical systems, provides powerful tools for dynamic decisions. Many of these tools fall into the category of *model-based control*. Namely, the methods rely on a mathematical model of the dynamical system, for example a state-space description or transfer function, to determine input laws, also referred to as *strategies*, with the aim of providing guarantees regarding properties of the dynamical system under the action of these strategies. For instance, the properties to be shaped include stability of an equilibrium point of the system, performance with respect to a control objective such as minimising actuation effort or tracking a desired trajectory, and robustness to external disturbances. As modern engineering systems are becoming increasingly complex due to technological advances in areas including microprocessors and memory, digital technologies, artificial intelligence, and robotics, deriving accurate system models becomes increasingly challenging. Examples include modern robotic systems such as snake-like robots, which due to their excellent mobility achieved by mimicking the movements of biological snakes show potential for a wide range of applications in challenging environments, including for search and rescue missions, for inspection and maintenance tasks and for space exploration. However, the locomotion of the underactuated snake-like robot relies on the complex friction forces between robot and ground, which are difficult to model accurately [3]. Another example are modern “smart grid” power systems, which are required to cope with ever increasing demand while receiving intermittent supply from renewable energy sources such as solar and wind power [4, 5]. The increasing use of digital tools in this context also opens new challenges such as preventing cyber-attacks to power grid control systems [6]. Even if an accurate model of such complex systems is available, it may be too complex and hence not well suited for control design [7]. While not a new topic in control theory, *data-driven control* methods, which aim to learn or replace system models by using measured data, have recently become increasingly popular, due to advances in sensing capabilities and computational resources [8]. A central question in the context of data-driven control is how to substitute a system model with data, while providing similar stability, performance and robustness guarantees to the ones characteristic of model-based control.

Classical control problems involve a single decision maker designing decision laws for the inputs to a dynamical system. With systems becoming increasingly interconnected and influenced by the inputs of various decision makers, new challenges arise. *Dynamic game theory* [9], a sub-field of game theory, is concerned with the dynamic interaction of strategic decision makers, also called *players*. Each player aims to optimise a performance criterion via the choice of decision law for the player’s action or input. The players’ performance criteria can involve both team-based and possibly conflicting individual objectives and hence may or may not be such that the players are in direct competition. Thus, dynamic game theory finds application in wide range of interaction scenarios. For example, consider a monetary union of several countries with a common central bank, such as the Eurozone, the interaction of the national fiscal policies and the monetary policy of the central bank can be studied as a dynamic game [10]. Further examples of dynamic decision-making problems which can be modelled as a dynamic game include military strategy [11], and environmental and epidemic management [12, 13]. The dynamic game formulation also naturally captures control problems involving multiple decision makers influencing a dynamical system. In fact, certain classes of dynamic games can be interpreted as a generalisation of optimal control problems from the single-player case to the multi-player case [14]. This ranges from designing controllers to attenuate disturbances [15] or to trade off between performance and disturbance attenuation objectives [16], to control of cyber-physical systems [17], power systems [18, 19], robotic systems [20, 21, 22, 23, 24] or general multi-agent systems. In the context of the latter, tools from dynamic game theory can be harnessed to tackle problems such as collision avoidance [25, 26] and formation control [27] and have recently gained attention for distributed control design [28, 29], i.e. to design controllers for each agent based on limited locally available information with the aim of achieving a desired collective behaviour for the entire multi-agent system. As with all decisions, the information available to each decision maker plays a key role in the outcome of a game. For example, any hierarchy among the players or whether the players can measure the current state of the decision-making process or only know the initial state can affect the type of solution sought. In many scenarios which can be modelled as dynamic games it is possible or even likely that different and incomplete information is available to each player. For example, consider a robotic system interacting with a human operator for arm reaching movements. Such a scenario may arise in a manufacturing setting to support the lifting of heavy objects or in a rehabilitation setting to train a patient's motor skills. The interaction between the human operator and the contact robot can be modelled as a dynamic game. However, the contact robot cannot know the human operator's motor behaviour and performance criteria a priori [20, 21]. Even in the "classical" game formulation with complete information, i.e. all players know the full system dynamics and the performance objectives of themselves and all other players, dynamic games constitute multi-objective optimisation problems, which are generally not straightforward to solve. Multiple solution concepts exist, and even with focus on a specific type of solution, there may exist multiple such solutions with different outcomes, which are more or less favourable for different players [9]. A commonly considered solution concept is the *Nash equilibrium*. At such a solution, the players' strategies are in equilibrium in the sense that no player has an incentive to deviate from the solution strategy while the other players' strategies remain unchanged, since such a deviation would incur a higher cost. Finding feedback strategies which satisfy this criterion is generally challenging. Notably, for games involving nonlinear dynamics, closed-form solutions do not generally exist. Even in the linear quadratic case - which in the single-player optimal control setting is well understood and can be solved using readily available techniques - multiple solutions may exist in the multi-player setting, and computing such equilibrium solutions involves the solution of coupled matrix equations which are generally challenging to solve [9, 30]. While feedback Nash equilibrium solutions and their computation have been extensively studied for certain classes of linear quadratic dynamic games, many interesting questions remain.

As the title suggests, the objective of this thesis is to study and develop game theoretic and data-driven methods for dynamic decisions, considering scenarios ranging from complete to incomplete information. Towards this objective, the following three sub-objectives are tackled.

1. Considering multi-player dynamic decision problems which can be cast as a class of linear quadratic dynamic games and are generally difficult to solve, with complete information, study feedback Nash equilibrium solutions and propose novel approximate solution concepts and iterative solution methods.

2. Considering control problems which are well understood in the complete information context, including stabilisation, optimal control and robust control of linear systems, in the context of incomplete information, namely if the system dynamics and/or the cost parameters are unknown to the decision maker, propose methods to solve the problems utilising data by extending recent data-driven control design methods.

3. Combining results from the previous two points, propose data-driven approaches to solve certain classes of linear quadratic dynamics games with incomplete information.

How this thesis addresses these three objectives is outlined in more detail in the following section.

1.2 Organisation and overview of contributions

The thesis is organised as illustrated in Figure 1.1. The main contributions of the following chapters are summarised below.

Chapter 2: Before the main results of this thesis are presented in Chapters 3-5, a review of the relevant literature and an introduction to important background material are provided in Chapter 2. To this end, Section 2.1 focuses on dynamic game theory, before Section 2.2 dives into preliminaries on data-driven control.

Chapter 3: In the first main results chapter, the focus lies on dynamic game theory, and in particular deterministic, non-cooperative, nonzero-sum, infinite-horizon, linear quadratic, discrete-time dynamic games and their feedback Nash equilibrium solutions. The contribution of this chapter is threefold. First, the study of the special case in which the dynamics of the game are described by scalar variables in Section 3.2 gives insights into the possible number and properties of different solutions. This is achieved by proposing a graphical representation of the conditions characterising feedback

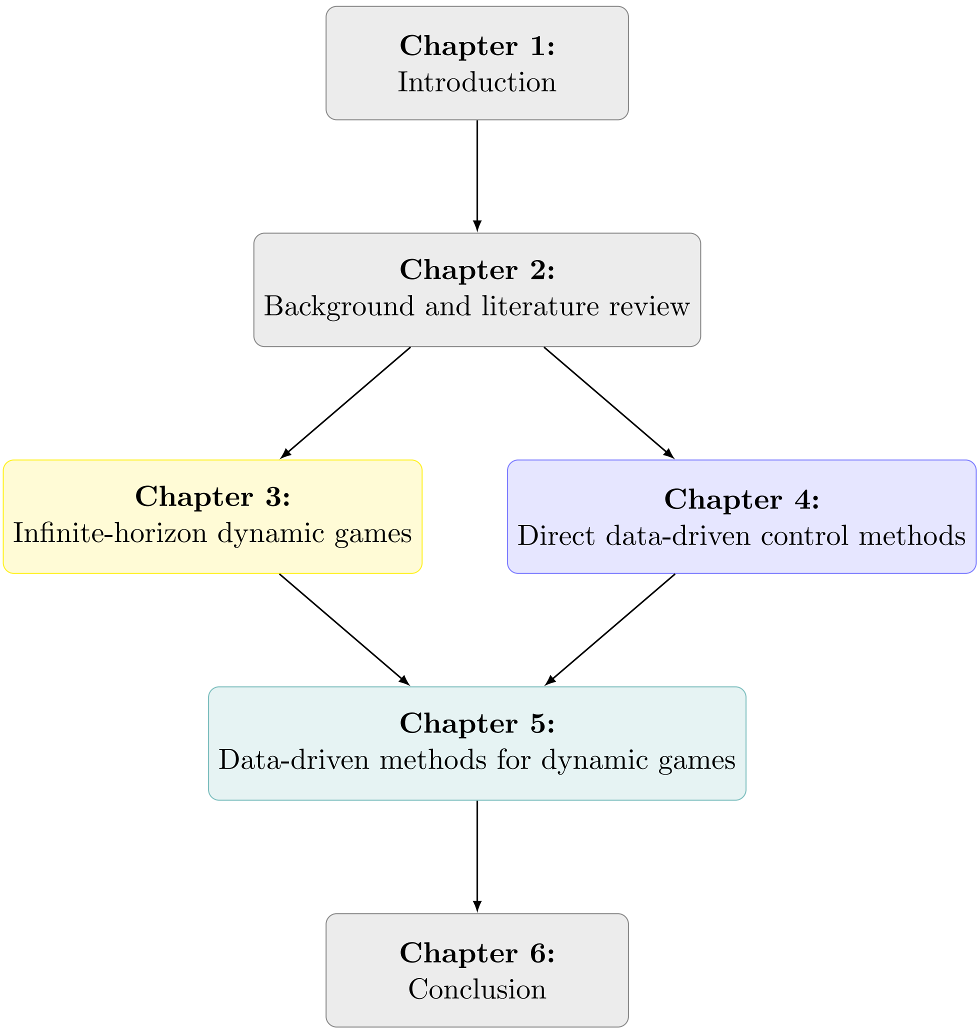


Figure 1.1: Organisation of the thesis.

Nash equilibria for the considered class of games. Via geometric arguments conditions in terms of the model and cost parameters are derived under which the game admits a certain number of feedback Nash equilibrium solutions and under which they possess certain properties. For scalar games involving two players, an alternative graphical representation is derived, which allows to confirm the results from a different perspective and provides additional insights. The results are illustrated via numerical examples. Second, shifting the focus back from scalar games to games involving general linear dynamics, a notion of approximate feedback Nash equilibrium is proposed in Section 3.3. The solution concept provides guarantees on the rate of convergence of the trajectories of the resulting closed-loop system. Its characterisation via matrix inequalities, the degree of approximation and the computation of solutions are discussed. The results are demonstrated via a macroeconomic policy design example in simulation. Third, four iterative methods to determine feedback Nash equilibrium solutions of the considered class of games are proposed and discussed in Section 3.4. The methods rely on the iterative solution of uncoupled matrix equations, more precisely Lyaponov or Riccati equations, to solve the set of coupled matrix equations associated with feedback Nash equilibrium solutions. Local convergence criteria are provided. The efficacy of the presented algorithms is demonstrated and compared to alternative algorithms by means of two illustrative numerical examples.

Chapter 4: The second main results chapter deals with direct data-driven control and features two main contributions. First, in Section 4.1 a recent data-driven control framework, which allows to design feedback controllers directly using data via the solution of convex optimisation problems by representing both the controller and the resulting closed-loop system with a comparatively small amount of measured data, is extended to the class of linear time-varying systems. Methods to design controllers with trajectory boundedness and performance guarantees for unknown linear time-varying systems based purely on an ensemble of input-state data, and without explicitly identifying the system dynamics, are proposed and discussed. No prior knowledge regarding the underlying time-variation is required. However, it is shown how such information can be utilised to relax the data requirements and derive infinite-horizon results from finite-length data for periodically time-varying systems. Both noise-free systems and systems affected by both measurement and process noise are considered. To demonstrate the efficacy and relevance of the results, both illustrative numerical examples and two practically motivated examples involving a voltage source converter and a snake-like robot are provided. Second, motivated by the fact that in many practical settings performance criteria are often not obvious a priori, linear quadratic optimal control problems with unknown cost functions are considered. In Section 4.2, a method to represent the cost function in a similar manner to the system dynamics using non-optimal finite-length data of the response of the state and a performance variable to exploring inputs is proposed. It is shown that in combination with the data-driven system representation this cost representation allows to solve optimal control problems with both unknown dynamics and unknown cost functions via purely data-dependent convex optimisation problems.

Chapter 5: In the final main results chapter, findings of Chapter 3 and Chapter 4 are combined to derive data-driven methods for dynamic games. The contribution is twofold. First, in Section 5.1 a class of infinite-horizon linear quadratic discrete-time dynamic games, in which different information is available to different players is considered. More precisely, assume one of the players does not know the cost functions which the other players are aiming to minimise and may not know the system dynamics. Hence, this player cannot determine a Nash equilibrium solution of the game using classical methods or using the dynamic game results of Chapter 3. It is shown that the data-driven control results of Section 4.2 are relevant for this class of games with asymmetric information structure and allow the "uninformed" player to compute a feedback Nash equilibrium strategy by compensating for the lack of information with measured data. The efficacy and relevance of the proposed results is demonstrated via a simulation example involving human-robot interaction. Second, in Section 5.2 the focus shifts to games in which not just one of the players is faced with incomplete information, but all players in the infinite-horizon linear quadratic discrete-time dynamic game have limited information available to them. Namely, each player only knows the own performance objective, but not the cost functions which the other players are aiming to minimise. Additionally, the players may not know the system dynamics. Data-driven versions of the algorithms in Section 3.4 are proposed to overcome this lack of information. By utilising measured data, in a similar way as proposed in Section 5.1, in the context of the iterative update laws from Section 3.4, it is shown in Section 5.2 that the players are able to jointly converge to a Nash equilibrium solution via scheduled experiments by taking turns to collect data to update their strategies. The performance of the data-driven algorithms is demonstrated and discussed via two illustrative numerical examples, before the practically motivated human-robot interaction example from Section 5.1 is revisited.

Chapter 6: The final chapter rounds off with a summary of the work and provides concluding remarks.

1.3 Published results

Results presented in this thesis have appeared in the following published works, as well as the co-authored papers [31], [32], as outlined in Table 1.1.

[BN1] B. Nortmann, A. Monti, M. Sassano, and T. Mylvaganam, "Nash equilibria for linear quadratic discrete-time dynamic games via iterative and data-driven algorithms," *IEEE Transactions on Automatic Control*, 2024.

[BN2] B. Nortmann and T. Mylvaganam, "Direct data-driven control of linear time-varying systems," *IEEE Transactions on Automatic Control*, vol. 68, no. 8, pp. 4888–4895, 2023.

[BN3] B. Nortmann, A. Monti, M. Sassano, and T. Mylvaganam, Feedback Nash equilibria for scalar two-player linear-quadratic discrete-time dynamic games," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 1772–1777, 2023.

[BN4] B. Nortmann and T. Mylvaganam, "Approximate Nash equilibria for discrete-time linear quadratic dynamic games," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 1760–1765, 2023.

[BN5] B. Nortmann, A. Monti, T. Mylvaganam, and M. Sassano, "Nash equilibria for scalar LQ games: iterative and data-driven algorithms," in *IEEE Conference on Decision and Control*, 2022, pp. 3801–3806.

[BN6] B. Nortmann and T. Mylvaganam, "Data-driven cost representation for optimal control and its relevance to a class of asymmetric linear quadratic dynamic games," in *European Control Conference*, 2022, pp. 2185–2190.

[BN7] B. Nortmann and T. Mylvaganam, "Data-driven control of linear time-varying systems," in *IEEE Conference on Decision and Control*, 2020, pp. 3939–3944.

Table 1.1: Overview of published results.

Chapter 3	Section 3.1	Two-player versions of the proofs of Theorem 3.1.1 and Corollary 3.1.1 also appear in [31].
	Section 3.2	Preliminary results (the two-player case in Section 3.2.4 and the two player examples in Section 3.2.5) are published in [BN3].
	Section 3.3	Published in [BN4] apart from Section 3.3.4.
	Section 3.4	Published in [BN1], preliminary results for scalar games are published in [BN5].
Chapter 4	Section 4.1	Noise-free results in Section 4.1.2 are published in [BN2], [BN7]. Robust control results in Scection 4.1.3 and results for periodically time-varying systems in Section 4.1.5 are published in [BN2]. The numerical LQR example in Section 4.1.6 is published in [BN7]. The snake-like robot example in Section 4.1.6 appears in [32].
	Section 4.2	Published in [BN6].
Chapter 5	Section 5.1	Published in [BN6] apart from the proof of Corollary 5.1.1.
	Section 5.2	Published in [BN1], preliminary results for scalar games are published in [BN5].

1.4 Notation

The sets of complex numbers, real numbers, integers and natural numbers are denoted by $\mathbb{C}$, $\mathbb{R}$, $\mathbb{Z}$ and $\mathbb{N}$, respectively. The zero matrix of appropriate dimension is denoted by 0 and the $n \times n$ identity matrix by I_n. Given a scalar $\lambda \in \mathbb{C}$, $|\lambda|$ denotes the modulus of λ. Given a vector $v \in \mathbb{R}^n$, $\|v\|$ denotes its Euclidean norm and given a matrix $M \in \mathbb{R}^{m\times n}$, $\|M\|$ denotes the induced 2-norm of M. The transpose of a vector (matrix) is denoted by $v^\top$ ($M^\top$) and the conjugate transpose by v^H (M^H). The vectorisation of M is denoted by $\mathrm{vec}(M)$. The block diagonal stacking of matrices M and N is written as $\mathrm{diag}\,(M, N)$. Given a square matrix A, $\mathbf{Tr}(A)$ denotes its trace, and $A \succ 0$ ($A \succeq 0$) denotes that A is positive definite (positive semi-definite). The spectral radius of A is denoted by $\rho(A)$. If A is invertible, A^{-1} denotes its inverse. Given a matrix B of full row rank, $B^\dagger$ denotes its right inverse. In matrix inequalities $\star$ denotes blocks (or matrices), which can be inferred by symmetry. Given a signal $z : \mathbb{Z} \to \mathbb{R}^\sigma$ the sequence $\{z(k), \ldots, z(k+T)\}$ is denoted by $z_{[k,k+T]}$ with $k, T \in \mathbb{Z}$ and we denote $|z|_k = \sup\{\|z(j)\|, 0 \le j \le k\} \le \infty$. The space of square-summable sequences is denoted by ℓ_2. A function $\gamma : \mathbb{R}_{\ge 0} \to \mathbb{R}_{\ge 0}$ is a class $\mathcal{K}$-function if it is continuous, strictly increasing and $\gamma(0) = 0$. Throughout the thesis the superscript * refers to an exact solution and the superscript * indicates an approximate solution.

Chapter 2

Background and literature review

Modern life heavily depends on complex systems, from the aeroplanes bringing us to destinations across the world and the satellite constellations enabling satellite navigation to advanced robots manufacturing our cars and smart power grids seamlessly integrating renewable energy sources. We take it for granted that such systems behave as expected, respond appropriately to our inputs, or even operate safely autonomously. However, many natural and engineered systems are governed by dynamics which may be complex and may exhibit unexpected and potentially dangerous behaviours. By providing tools to analyse systems and to design and implement laws governing dynamic decisions, control theory allows us to understand and modify the behaviour of dynamical systems. However, with systems becoming increasingly complex, intelligent, digital and interconnected due to significant advances in microprocessors and memory, digital technologies, artificial intelligence and robotics over the last forty years, new challenges have been arising in the context of control design. A central question is how to design controllers guaranteeing stability and performance in the face of uncertainty or limited information. This is particularly true for systems influenced by the inputs of multiple decision makers. This thesis addresses this question by combining tools from the fields of game theory and data-driven control design. The following two sections aim to summarise the background and state-of-the-art in these two fields to put the results of the following chapters into perspective.

2.1 Dynamic game theory

Collisions of interest predate humankind and have been ubiquitous throughout our history. Conflict situations have shaped our societies, have provided motifs for art and literature and have fuelled technological advances. Conflict arises when interacting individuals or parties pursue their own individual interest, which may at least partly be clashing with the others' interests. Each individual or party has to make decisions from a set of options, where each possible decision will likely lead to a different outcome. However, this outcome also depends on the decisions taken by the other parties involved, and different possible outcomes may be valued differently by the different decision makers [9]. While thoughts about and insights into this type of problem date back to ancient times, appearing for example in texts of Plato [33], a systematic mathematical framework to model and analyse conflict situations, or more generally multi-person or multi-party decision making (which does not necessarily involve direct conflict), has been pioneered in the 1930 and 1940 by von Neumann and Morgenstern [34] and is called *game theory*.

Solution concepts in game theory

In contrast to single-person or single-party decision making, where the concept of optimality is unambiguous, the "best" solution is equivocal if multiple decision makers are involved. Hence, game theory concerns the design of *strategies* (decision rules) based on which the rational decision makers - referred to as *players* - take their *actions* to achieve a certain outcome. This type of problem is called a *game*. A classical case is the two-player *zero-sum* game, which is defined by a single performance criterion, with one player aiming to minimise it and the other player aiming to maximise it via their respective choices of actions. What one player gains the other loses, or in other words, their objective functions sum up to zero. An example is the rope-pulling or "tug of war" game, in which two players (which can be individuals or teams) pull on opposite ends of a rope with the aim of moving the rope and the other player a certain distance in one direction. If the two opponents are equally strong, the most favourable strategy for each player is clearly to pull the rope as hard as they can in the opposite direction to the other player and the rope does not move. The strategies are in *equilibrium*, that is, each player's strategy is optimal against the other's strategy. Such a solution is known as a *saddle-point solution* [9, 35].

The solution of a game becomes less straightforward if there are $N > 2$ players involved or if each player is trying to optimise their own performance criterion, and the game is such that the players' performance criteria do not add up to zero, i.e. *nonzero-sum* games. To illustrate this, consider the well-known prisoners' dilemma (see e.g. [35]). Two prisoners are awaiting their trial. If found guilty, they will face a 10 year prison sentence. The prisoners are held separately and each offered the chance to testify and give evidence incriminating the other prisoner. If only one of the prisoners testifies, the other prisoner will be found guilty and the testifying prisoner will be set free. If both choose to testify, each will receive a reduced 5 year sentence. If neither prisoner testifies, they can both only be convicted of a lesser crime with a shorter sentence of 2 years. This results in the following bimatrix game

$$C_1 = \begin{array}{c}\text{Player 2}\\ \begin{array}{|c|c|c|} \hline nt & t & \\ \hline 2 & 10 & nt \\ \hline 0 & 5 & t \\ \hline \end{array}\end{array} \text{Player 1,} \qquad C_2 = \begin{array}{c}\text{Player 2}\\ \begin{array}{|c|c|c|} \hline nt & t & \\ \hline 2 & 0 & nt \\ \hline 10 & 5 & t \\ \hline \end{array}\end{array} \text{Player 1.} \tag{2.1}$$

The matrix C_1 (C_2) shows the cost incurred by Player 1 (Player 2) for the possible combinations of the actions taken by the two players, where each player can choose between the options nt (no testimony) and t (testimony).

Does there exist an equilibrium solution with similar properties as the one observed in the rope-pulling example? If both players play t, then neither player can reduce their cost by unilaterally changing strategy, i.e. $C_1(t, t) \leq C_1(nt, t)$ and $C_2(t, t) \leq C_2(t, nt)$. The pair of strategies (t, t) is called a *Nash equilibrium* solution (named after John Nash who introduced the concept in [36], [37]) of the game (2.1). Note that, in general, a bimatrix game may admit multiple Nash equilibrium solutions with different outcomes. If that is the case, the outcome of one Nash equilibrium solution may be more favourable than the outcome of another Nash equilibrium solution for one or both of the players.

Despite the solution (t, t) being the unique Nash equilibrium of the game (2.1), it is clearly not an optimal solution. A better outcome for both players is achieved if they both choose not to testify, i.e. by playing the pair of strategies (nt, nt). Note that this pair of strategies is not an equilibrium, since each player could benefit from changing their strategy. As such, this solution is vulnerable to cheating. However, this highlights that the players could benefit from cooperating by agreeing to both play nt. The solution (nt, nt) is such that the cost for each of the players cannot be improved simultaneously, i.e. there exists no other solution such that at least one player achieves a lower cost and no other player is worse off. This is called a *pareto-optimal* solution [38, 9, 35].

In a non-cooperative setting, another solution of interest arises if the players choose their strategy to minimise their cost against the worst possible strategies the opponents could choose. As such, the players effectively each play a two-player zero-sum game, aiming to minimise their own cost assuming all other players collectively aim to maximise it, and do not take the performance criteria of the other players into account. This *mimimax* solution [9, 35] may be a pessimistic choice, however, it gives the players a level of security if they are not sure how the rivals will select their strategies. This may be relevant if there are multiple equilibria which are more or less favourable for different players or if a player is not aware of the other players' performance criteria. In the game (2.1), Player 1 can mitigate against the most harmful action of Player 2 (t) by choosing t as well. Similarly, Player 2 can alleviate the effect of the most detrimental action of Player 1 (which is also t) by playing t. Hence, the pair of strategies (t, t) constitutes a minimax solution. Note that in this special case, the minimax solution coincides with the unique Nash equilibrium solution of the game and is a natural choice if there is no cooperation between the players. However, in general, a minimax outcome is worse or equally good, but not better than any Nash equilibrium outcome.

Nash equilibrium, pareto-optimal and minimax solutions have in common that they are relevant if the roles of the players are symmetric. A natural question is then what a favourable solution looks like if there is a hierarchy among the players in the decision process. In the game (2.1), let Player 1 be the *leader*, who announces the chosen strategy first, and let Player 2 be the *follower*, who reacts rationally to the leader's decision. Hence, Player 1 has to take the possible responses of Player 2 into account when choosing the most favourable strategy. If Player 1 chooses nt, the best response for Player 2 is t, resulting in the cost $C_1(nt, t) = 10$. If Player 1 instead plays t, the best response for Player 2 is still t, giving $C_1(t, t) = 5$. Hence, the latter is the most favourable choice for Player 1 in this hierarchical decision process. The solution (t, t) is called the *Stackelberg* equilibrium solution [39] of the game (2.1) with Player 1 as the leader. While in this example the Stackelberg equilibrium coincides with the Nash equilibrium and the minimax solution, this is not always the case. In fact, a Stackelberg equilibrium outcome may be more favourable for one or both of the players compared to the other solution outcomes.

Static versus dynamic, information structures and cooperation

The above examples highlight the importance of the *role of information* available to each player in the solution of games. The most appropriate strategy for each player depends on the knowledge of the objectives of the other players, on whether the other players are happy to cooperate and can be trusted, and on the hierarchy among the players. The aforementioned examples are both *static games*, in the sense that the players only act once, and the outcome of the interaction depends on that single instantaneous decision. A game is *dynamic* if the interaction evolves over time, if the order in which a player makes decisions is important and if the players can use strategies which depend on previous actions. While illustrated by means of static examples, the solution concepts introduced above also extend to the dynamic case. In *dynamic games*, the players' strategy choices are further influenced by whether the underlying information structure is *feedback*, that is, each player knows exactly to which *state* the game has evolved at each point in time, or *open-loop*, in which case only the initial state is known. The prisoner's dilemma example (2.1) highlights how cooperation can lead to a better outcome for the players involved. Games (both static and dynamic), which allow players to work together to their advantage or form coalitions are called *cooperative games*.

A survey of dynamic cooperative games is provided in [40]. In the commonly considered setting in which all players cooperate to achieve their objectives by communicating and entering into binding agreements, without side-payments taking place (see e.g. [30]), the multi-objective game problem can be reduced to a single-objective problem with the performance criterion being a weighted sum of the individual performance criteria of the players. This single-objective optimisation problem leads to a pareto-optimal solution parameterised with respect to the weighting parameters. In the dynamic game setting this single-objective problem can be solved using tools from optimal control theory. In general, many different pareto-optimal solutions may exist for a given game. While by definition any pareto-optimal solution is such that the outcomes for all players cannot be improved simultaneously, different pareto-optimal solutions may be more or less attractive for certain players. Hence, the question arises as to which is the "best" solution for the players (with potentially conflicting interests) to agree on. Bargaining theory [41] provides approaches to answer this question by comparing the benefits, which the players can gain by cooperating, to the solution of the corresponding non-cooperative problem. A game is a *non-cooperative game*, if the players pursue their own individual, often partly conflicting interests without collaboration. This is the type of games considered in this thesis. A comprehensive analysis of dynamic non-cooperative games is provided in [9].

Differential games and discrete-time dynamic games

The focus of this thesis is on a class of dynamic games in which the interaction is described by the evolution of a dynamical system influenced by the inputs of the $N \in \mathbb{N}$ players. In reference to the type of dynamic equation involved, these dynamic games are also known as *differential games* in a continuous-time context [11, 35] and as *difference games* (or simply *discrete-time dynamic games*, which is how they are referred to throughout this thesis) in a discrete-time context [9]. Namely, consider the dynamical system

$$\dot{x} = f^c(x, u_1, \ldots, u_N), \tag{2.2}$$

with $f^c : \mathbb{R}^n \times \mathbb{R}^{m_1} \times \ldots \times \mathbb{R}^{m_N} \to \mathbb{R}^n$ such that $f^c(0, 0, \ldots, 0) = 0$ is an equilibrium point, where $x \in \mathbb{R}^n$ denotes the state of the system and $u_i \in \mathbb{R}^{m_i}$ is the input or action of player i, for $i = 1, \ldots, N$. Under the assumption that the players are rational, let each player i aim to minimise a cost functional

$$J_i\left(x(0), u_1(\cdot), \ldots, u_N(\cdot)\right) = g_i^c\left(x(T)\right) + \int_0^T l_i^c\left(x(\tau), u_1(\tau), \ldots, u_N(\tau)\right) d\tau, \tag{2.3}$$

over the time interval $t \in [0, T]$, where $g_i^c : \mathbb{R}^n \to \mathbb{R}$ represents a terminal cost and $l_i^c : \mathbb{R}^n \times \mathbb{R}^{m_1} \times \ldots \times \mathbb{R}^{m_N} \to \mathbb{R}$ represents a running cost, via the choice of control strategy for the action u_i, for $i = 1, \ldots, N$. The dynamics (2.2) and cost functionals (2.3), $i = 1, \ldots, N$, describe a (*finite-horizon*) differential game. If the asymptotic behaviour of the interaction of the players is of interest, i.e. the case in which $T \to \infty$, the performance criterion becomes

$$J_i\left(x(0), u_1(\cdot), \ldots, u_N(\cdot)\right) = \int_0^\infty l_i^c\left(x(\tau), u_1(\tau), \ldots, u_N(\tau)\right) d\tau, \tag{2.4}$$

and the game (2.2), (2.4), $i = 1, \ldots, N$, is an *infinite-horizon* differential game. Similarly, consider the discrete-time dynamical system

$$x(k+1) = f^d\left(x(k), u_1(k), \ldots, u_N(k)\right), \tag{2.5}$$

for $k \in \mathbb{Z}$, with $f^d : \mathbb{R}^n \times \mathbb{R}^{m_1} \times \ldots \times \mathbb{R}^{m_N} \to \mathbb{R}^n$ such that $f^d(0, 0, \ldots, 0) = 0$ and the cost functional

$$J_i\left(x(0), u_1(\cdot), \ldots, u_N(\cdot)\right) = g_i^d\left(x(N_f)\right) + \sum_{k=0}^{N_f-1} l_i^d\left(x(k), u_1(k), \ldots, u_N(k)\right), \tag{2.6}$$

with $g_i^d : \mathbb{R}^n \to \mathbb{R}$ and $l_i^d : \mathbb{R}^n \times \mathbb{R}^{m_1} \times \ldots \times \mathbb{R}^{m_N} \to \mathbb{R}$, which player i aims to minimise via the choice of control strategy for the action u_i over the time interval $k = 0, \ldots, N_f$, for $i = 1, \ldots, N$. The dynamics (2.5) and cost functionals (2.6), $i = 1, \ldots, N$, describe a (finite-horizon) discrete-time dynamic game. Analogous to the continuous-time case, in the limit as $N_f \to \infty$, the terminal cost is zero and the performance criterion becomes

$$J_i\left(x(0), u_1(\cdot), \ldots, u_N(\cdot)\right) = \sum_{k=0}^{\infty} l_i^d\left(x(k), u_1(k), \ldots, u_N(k)\right), \tag{2.7}$$

and the game (2.5), (2.7), $i = 1, \ldots, N$, is an infinite-horizon discrete-time dynamic game. In the infinite-horizon context (both in continuous-time and discrete-time) it is common to restrict the search for solution strategies to stabilising strategies, which ensure the value of the cost functionals (2.4) or (2.7), for $i = 1, \ldots, N$, remains finite.

Definition 2.1.1. A set of strategies $\{\phi_1(\cdot), \ldots, \phi_N(\cdot)\}$ is *admissible* if the set of control actions $\{u_1, \ldots, u_N\}$, with $u_i = \phi_i(\cdot)$ for $i = 1, \ldots, N$, renders the zero equilibrium of the continuous-time system (2.2) or the discrete-time system (2.5) (locally) asymptotically stable.

Applications and relevance

The defined classes of dynamic games are applicable to model interactions which may, or may not, be competitive in a variety of fields, with applications ranging from ecology [42, 43, 44, 12], economics [45, 44, 12], epidemiology [46, 13], robotics [20, 21, 47, 22, 23, 26] and power systems [18, 19] to politics [48], military strategy [11] and cyber-security [17]. Differential games and discrete-time dynamic games can be considered as a generalisation of optimal control problems to the multi-player case [14, 9]. Hence, the classes of games apply naturally to various problems arising in control theory. For instance, worst-case design problems, such as the H_∞-control problem, can be formulated and solved as a zero-sum dynamic game [15]. Nonzero-sum dynamic games, on the other hand, can capture the mixed H_2/H_∞-control problem [16, 49] and have found applications, for instance, in multi-agent collision avoidance [25, 23, 24, 26], formation control [27] and distributed control [28, 29].

Nash equilibria and associated challenges

As illustrated on the basis of the prisoner's dilemma example (2.1), Nash equilibrium solutions are of natural interest and hence a commonly considered solution concept in non-cooperative games, which are the focus of this thesis. For the considered classes of games, Nash equilibria are defined as follows.

Definition 2.1.2. A set of strategies[1] $\{\phi_1^\star(\cdot),\dots,\phi_N^\star(\cdot)\}$ constitutes a Nash equilibrium solution of an N-player differential game or discrete-time dynamic game, if the inequality

$$J_i^\star = J_i\left(x(0),\phi_1^\star(\cdot),\dots,\phi_N^\star(\cdot)\right) \le J_i\left(x(0),\phi_1^\star(\cdot),\dots,\phi_{i-1}^\star(\cdot),\phi_i(\cdot),\phi_{i+1}^\star(\cdot),\dots,\phi_N^\star(\cdot)\right), \tag{2.8}$$

holds for all $\{\phi_1^\star(\cdot),\dots,\phi_{i-1}^\star(\cdot),\phi_i(\cdot),\phi_{i+1}^\star(\cdot),\dots,\phi_N^\star(\cdot)\}$, for $i=1,\dots N$. The strategy $\phi_i^\star(\cdot)$ is referred to as a Nash equilibrium strategy of player i, $i=1,\dots,N$, whereas the set $\{J_1^\star,\dots,J_N^\star\}$ is the corresponding Nash equilibrium outcome.

Under a feedback information structure, i.e. if $\phi_i(\cdot)=\phi_i(t,x(t))$ (continuous-time) or $\phi_i(\cdot)=\phi_i(k,x(k))$ (discrete-time), dynamic programming [50] arguments can be used to show that Nash equilibrium solutions of differential games are characterised by the solutions of a set of coupled partial differential equations (PDEs) and Nash equilibrium solutions of discrete-time dynamic games by the solutions of a set of coupled difference equations. More precisely, in the continuous-time context, consider the Hamiltonian associated with player i,

$$H_i\left(x,u_1,\dots,u_N,\frac{\partial V_i}{\partial x}\right) = l_i^c\left(x,u_1,\dots,u_N\right) + \left(\frac{\partial V_i}{\partial x}\right)^\top f^c(x,u_1,\dots,u_N), \tag{2.9}$$

where $V_i(t,x(t))$ is the *value function* of player i, namely, the infimum of the cost functional of player i over all permissible control strategies for u_i, such that $V_i(0,x(0))=J_i^\star$, for $i=1,\dots,N$. In the finite-horizon game (2.2), (2.3), $i=1,\dots,N$, the value function is also such that $V_i(T,x(T))=g_i^c\left(x(T)\right)$. In line with Definition 2.1.2, and according to Bellman's principle of optimality, V_i is the solution of the PDE

$$-\frac{\partial V_i}{\partial t} = \min_{u_i} H_i\left(x,\phi_1,\dots,\phi_{i-1},u_i,\phi_{i+1},\dots,\phi_N,\frac{\partial V_i}{\partial x}\right). \tag{2.10}$$

Since there are no constraints on the permissible strategies[2], the control action minimising the right

[1] In the infinite-horizon games (2.2), (2.4), $i=1,\dots,N$, (continuous-time) and (2.5), (2.4), $i=1,\dots,N$, (discrete-time), the search for a Nash euqilibrium solution is restricted to admissible strategies in the sense of Definition 2.1.1.

[2] Apart from admissibility in the sense of Definition 2.1.1 in the infinite-horizon game (2.2),(2.4), $i=1,\dots,N$.

hand side of (2.10), $u_i^\star = \phi_i^\star$, is such that

$$\frac{\partial H_i\left(x, \phi_1, \ldots, \phi_{i-1}, u_i, \phi_{i+1}, \ldots, \phi_N, \frac{\partial V_i}{\partial x}\right)}{\partial u_i} = 0.$$

Hence, a Nash equilibrium solution $\{\phi_1^\star, \ldots, \phi_N^\star\}$ satisfies the coupled PDEs

$$-\frac{\partial V_i}{\partial t} = H_i\left(x, \phi_1^\star, \ldots, \phi_N^\star, \frac{\partial V_i}{\partial x}\right), \tag{2.11}$$

for $i = 1, \ldots, N$, subject to the system dynamics (2.2). Note that in the infinite-horizon game (2.2), (2.4), $i = 1, \ldots, N$, $V_i(t, x(t)) = V_i(x(t))$, for $i = 1, \ldots, N$, hence the coupled PDEs become

$$0 = H_i\left(x, \phi_1^\star, \ldots, \phi_N^\star, \frac{\partial V_i}{\partial x}\right), \tag{2.12}$$

for $i = 1, \ldots, N$, subject to (2.2). Similarly, in the discrete-time context, recall Definition 2.1.2 and consider the functional equation of dynamic programming for player i,

$$V_i(k, x(k)) = \min_{u_i}\left(l_i^d\left(x(k), \phi_1(k), \ldots, \phi_{i-1}, u_i, \phi_{i+1}, \ldots, \phi_N(k)\right) + V_i(k+1, x(k+1))\right), \tag{2.13}$$

where the value function associated with player i, $V_i(k, x(k))$ is such that $V_i(0, x(0)) = J_i^\star$, and in the finite-horizon game (2.5), (2.6), $i = 1, \ldots, N$, $V_i(N_f, x(N_f)) = g_i(x(N_f))$. The right hand side of (2.13) is minimised by $u_i = \phi_i^\star$. Hence, a Nash equilibrium solution $\{\phi_1^\star, \ldots, \phi_N^\star\}$ satisfies the coupled difference equations

$$V_i(k, x(k)) = l_i^d\left(x(k), \phi_1^\star(k), \ldots, \phi_N^\star(k)\right) + V_i(k+1, x(k+1)), \tag{2.14}$$

for $i = 1, \ldots, N$, subject to the system dynamics (2.5). Note that in the infinite-horizon game (2.5), (2.7), $i = 1, \ldots, N$, $V_i(k, x(k)) = V_i(x(k))$, for $i = 1, \ldots, N$, hence the coupled difference equations become

$$V_i(x(k)) = l_i^d\left(x(k), \phi_1^\star(k), \ldots, \phi_N^\star(k)\right) + V_i(x(k+1)), \tag{2.15}$$

for $i = 1, \ldots, N$, subject to (2.5).

Feedback Nash equilibria of discrete-time dynamic games are the main focus of the game theoretic results in this thesis. As in optimal control problems (which are effectively a special case of the considered class of dynamic games with $N = 1$), an alternative approach to determining necessary conditions for a solution is via variational methods. This approach relying on Pontryagin's Maximum Principle [51] is a natural choice if *open-loop Nash equilibrium* solutions are considered.

Determining a solution, if any exists, to the coupled PDEs (2.11), (2.12) or the coupled difference equations (2.14), (2.15), $i = 1, \ldots, N$, associated with feedback Nash equilibrium solutions is generally challenging, particularly for games involving large systems influenced by many players [35, 52]. With the exception of certain special cases, closed-form solutions cannot readily be found. In this thesis, the focus lies on the class of *linear quadratic (LQ) dynamic games*, i.e. games defined by linear dynamics and quadratic cost functionals. In addition to their practical relevance (LQ dynamic games arise, for example, in many engineering and economics applications [30]), some analytical results can be derived for this class of games. Hence, LQ dynamic games, in particular the continuous-time case, i.e. LQ differential games, have been extensively studied in the literature [30, 9]. While reminiscent of the well-known linear quadratic regulator (LQR) problem for optimal control of linear systems influenced by the input of a single decision maker (see e.g. [52]), general multi-player LQ dynamic games are neither as well understood nor as easily solved as the single-player counterpart. For example, while the solution to the LQR problem is well known to be a static state-feedback law, multiplayer LQ dynamic games may admit nonlinear feedback strategies as Nash equilibrium solutions [53]. However, in this thesis, the attention is focused on linear state-feedback strategies, as is common in the context of LQ games (see e.g. [30, 9])[3]. This not only has the advantage of preserving the linearity of the system in closed-loop, but also results in the conditions characterising feedback Nash equiblibrium solutions, namely the coupled PDEs (2.11), (2.12) or the coupled difference equations (2.14), (2.15), $i = 1, \ldots, N$, reducing to coupled ordinary differential equations (ODEs) or simpler difference equations, respectively, in the finite-horizon case, or coupled algebraic equations in the infinite-horizon case. While this is computationally appealing, even in this simpler setting solving the coupled equations characterising feedback Nash equilibrium solutions is generally challenging and multiple solutions with different outcomes may exist [9, 35, 30]. Consequently, the existence, uniqueness and computation of feedback Nash equilibria in LQ dynamic games has been extensively studied, particularly in the continuous-time context, see e.g. [9, 54, 55, 56, 57, 58, 30]. The infinite-horizon, discrete-time case, however, has received less attention, see e.g. [9, 59]. Note that in this case, additional mixed product terms of the decision variables appear in the coupled algebraic equations associated with feedback Nash equilibria [59], which leads to additional challenges and complications in the solution of LQ discrete-time dynamic games compared to the continuous-time counterpart.

Iterative and approximate solution methods

Due to the difficulties of solving the coupled equations associated with feedback Nash equilibrium solutions of dynamic games, approximate or iterative solution methods are of interest. Approximate Nash equilibrium solution concepts for differential games have been introduced in [60, 61, 62]. A

[3] As common in the dynamic games literature, the search for equilibrium strategies is restricted to linear feedback strategies acting on the full state vector. Solutions in terms of strategies which act on a subset of states are beyond the scope of this thesis.

common notion of approximation is the ϵ-Nash equilibrium. A set of strategies[4] $\{\phi_1^*(\cdot), \ldots, \phi_N^*(\cdot)\}$ constitutes an ϵ-Nash equilibrium solution of an N-player dynamic game, if the inequality

$$J_i^* = J_i\left(x(0), \phi_1^*(\cdot), \ldots, \phi_N^*(\cdot)\right) \leq J_i\left(x(0), \phi_1^*(\cdot), \ldots, \phi_{i-1}^*(\cdot), \phi_i(\cdot), \phi_{i+1}^*(\cdot), \ldots, \phi_N^*(\cdot)\right) + \epsilon, \tag{2.16}$$

holds for all $\{\phi_1^*(\cdot), \ldots, \phi_{i-1}^*(\cdot), \phi_i(\cdot), \phi_{i+1}^*(\cdot), \ldots, \phi_N^*(\cdot)\}$, for $i = 1, \ldots N$, for some $\epsilon \geq 0$. The strategy $\phi_i^*(\cdot)$ is referred to as an ϵ-Nash equilibrium strategy of player i, $i = 1, \ldots, N$, whereas the set $\{J_1^*, \ldots, J_N^*\}$ is the corresponding ϵ-Nash equilibrium outcome. The additional term ϵ on the right hand side of (2.16) represents a degree of approximation. Depending on the problem formulation, the degree of approximation can be influenced by various factors, which leads to different variations of this notion of approximate Nash equilibrium [61, 62].

Even the computation of approximate Nash equilibria may be challenging. In practice, it can hence be of interest to numerically determine solutions. Algorithms to iteratively determine Nash equilibria have been suggested in [63, 64] for general nonlinear differential games and in [65, 30, 66, 67, 68, 69] for LQ differential games. Most algorithms are presented without a proof of convergence or convergence guarantees are limited to approximate equilibria or the special case in which there exists a unique feedback Nash equilibrium. Note that in general the number of feedback Nash equilibria can range from zero to infinity [30]. While the literature in the context of iterative solution methods for feedback Nash equilibria in dynamic games focuses mainly on the continuous-time setting, methods in the discrete-time setting include [70] for finite-horizon games, as well as [71] for nonlinear and [72, 73] for LQ discrete-time dynamic games in the context of reinforcement learning. Similar limitations on convergence guarantees as in the continuous-time case apply. However, in [74] a policy iteration algorithm is provided with conditions ensuring convergence to *a* Nash equilibrium.

Games with incomplete or local information, inverse games, and uncertainty

The discussion above, in particular in the context of the prisoners' dilemma example (2.1), highlights the role of information in game problems, including how it can influence the choice of solution concept. Even with focus on feedback Nash euqilibria of games it is important to specify the information available to each player. In the "classical" dynamic game formulation, each player has full knowledge of all system parameters and the performance criteria of all players [35]. However, in many settings, different and incomplete information may be available to each player.

On the one hand, in many dynamic games applications the system dynamics are complex and difficult to model. Algorithms to determine feedback Nash equilibrium solutions for games with *unknown*

[4] In the infinite-horizon games (2.2), (2.4), $i = 1, \ldots, N$, (continuous-time) and (2.5), (2.4), $i = 1, \ldots, N$, (discrete-time), the search for an approximate Nash euqilibrium solution is restricted to admissible strategies in the sense of Definition 2.1.1.

system dynamics have been proposed in [64, 75, 69] for the continuous-time case and [76, 72, 73, 74] for the discrete-time case.

On the other hand, in a competitive environment, a player is often unlikely to know in advance the full system or the objectives (performance criteria) of all the other players. Examples include multi-agent systems [29], cyber-physical systems [77] or human-robot systems [20]. In particular if systems are large-scale or comprise interconnected subsystems, scalability or communication constraints can make a centralised solution of the game infeasible. In [29] approximate Nash equilibria of distributed differential games, in which each player only communicates with its neighbours via a directed graph and hence only has access to limited local information, are determined via (fictitious) local differential games. Reinforcement learning algorithms for similar graphical games, in which only local information is available to each player, are introduced in [78] in a discrete-time setting and in [79] in a continuous-time setting. A distributed learning algorithm to compute Nash equilibrium strategies for the control of networked systems is proposed in [80]. If performance criteria are unknown, but the system dynamics are available and "expert data" corresponding to a solution of the game, such as equilibrium strategies or trajectories, can be collected, then *inverse dynamic game* methods (see e.g. [81] and references therein) can be used to learn or reconstruct the performance criteria of some or all players.

The class of dynamic games considered in this thesis and defined above is deterministic, i.e. the dynamics (2.2) or (2.5) governing the evolution of the interaction between the players and their environment do not involve randomness. A common and powerful approach to model system uncertainties and handle random noise and disturbances is by modelling the dynamics as a stochastic process [82]. Games involving such dynamics are known as stochastic dynamic games (for an overview see e.g. [83, 30, 9]). This class of games also naturally lends itself to the study of decentralised, distributed or local computation of solutions [84, 85, 86, 87][5]. In this context, team theory problems (see e.g. [90] and references therein) should be mentioned. Team problems concern multiple decision makers influencing a stochastic process with the aim of optimising a common performance criterion, however, each decision maker has access to different information regarding the underlying uncertainties. Team theory hence has similarities with cooperative stochastic game theory [40]. Stochastic counterparts to the class of games considered in this thesis are analysed in [9, 30]. Under the standard assumption that the disturbances and the initial states are independent Gaussian random vectors and that the disturbances are white noise with zero mean, the conditions characterising feedback Nash equilibria in the LQ setting coincide with those of the corresponding deterministic problem [30]. This highlights the relevance of studying the number, properties and computation of feedback Nash equilibrium solutions of deterministic LQ dynamic games.

[5] The references listed here and in the paragraph above in the context of decentralised or distributed control are examples of the use of game theoretic methods (involving games similar to the class of games considered in this thesis) in this context. For a comprehensive overview of decentralised control see e.g. [88, 89].

In this thesis, the focus lies on feedback Nash equilibrium solutions to deterministic, non-cooperative, nonzero-sum, infinite-horizon, LQ, discrete-time dynamic games. Despite being of practical relevance in many applications ranging from economics to engineering and control, this class of games has up to now received limited attention in the literature compared to its continuous-time counterpart. In Chapter 3, Section 3.2 starts with the study of scalar games, i.e. games involving scalar dynamics and scalar inputs, via geometric approaches to determine conditions under which a game admits a certain number of feedback Nash equilibria. This analysis allows to establish some intuition regarding the existence and uniqueness of solutions of the considered class of games. In the following sections, the focus shifts back to games involving general linear dynamics. Motivated by the challenges of computing feedback Nash equilibrium solutions, a notion of approximate feedback Nash equilibrium with guarantees on the rate of convergence of the trajectories of the resulting closed-loop system is introduced in Section 3.3, and iterative Nash equilibrium finding algorithms are proposed and analysed in Section 3.4. While the results in Chapter 3 rely on full system and cost information, Chapter 5 deals with the solution of games under partial information. By combining the results in Chapter 3 with data-driven methods, results for games with asymmetric information are presented in Section 5.1 and data-driven Nash equilibrium-finding algorithms are introduced in Section 5.2.

2.2 Data-driven control

As humans we are constantly learning. Whether intentionally or inadvertently, through practice, from our past experiences, through the observation of others or from literature and historical records - the interaction with our environment shapes our behaviour and decisions. The idea of machines learning from their environment in a similar way is at the core of to the notion of artificial intelligence, which has received increasing attention since the 1940s. Today it plays an ever-expanding role in our lives, from recommending us which YouTube videos to watch to opening up new possibilities (and challenges) in engineering and robotics, thanks to improving computational capacities allowing to collect, store and process large amounts of data.

Learning from data for dynamic decisions

Mathematical models are a key tool to analyse and influence systems, and as such form the basis of *model-based control*, which provides many powerful methods for dynamic decision making. With modern engineered systems becoming increasingly complex, obtaining accurate system models is increasingly challenging, if not impossible, and models derived from first principles may be too complex for control design. However, modern processes generate and store large amounts of data. Consequently, various methods have been developed to exploit this vast resource and use measured data - in different ways and with different objectives - to recover or replace system information. Examples include system identification [91, 92], adaptive control [93] and learning control [94]. Many data-based approaches focus on identifying or updating a system model, which can then be used in a second step to design a control law using any "classical", model-based technique. Hence, such methods are commonly referred to as *indirect* data-driven methods. In contrast, *direct data-driven control* methods aim to control a system directly using measured data, without explicitly identifying a system model [95]. This is not a new concept in control theory, but can be traced back to the work on proportional-integral-derivative (PID) controller tuning by Ziegler and Nichols [96] in the 1940s. Further earlier contributions to direct data-driven control include unfalsified control [97], iterative feedback tuning [98], virtual reference feedback tuning [99] and model-free adaptive control [100]. For more references, see e.g. [8, 101]. In addition to their theoretical value, direct data-driven control methods are also attractive for situations in which system identification can be difficult, expensive or time-consuming, and the resulting models might not be suitable for control design [7]. Moreover, direct data-driven analysis and control may be feasible even when unique system identification is not [102]. The topic has recently attracted significant attention. In particular, with the availability of increasing computational power and novel machine learning techniques, direct data-driven controllers using neural networks and reinforcement learning [103, 104, 105] have gained interest.

Direct data-driven control via Willems et al's fundamental lemma

A central question in direct data-driven control is how to substitute a system model with data. For linear time-invariant (LTI) systems, a recent line of research addresses this question via Willems et al.'s *fundamental lemma* [106]. In brief, the result states that all possible trajectories an LTI system can produce can be parameterised by a single, finite-length input-output trajectory - provided the input sequence sufficiently excites the system dynamics. As such, it naturally lends itself to address analysis and control problems from a data-driven perspective. This has been harnessed first in the behavioural framework (see e.g. [107, 108, 109]), in which the fundamental lemma was first established, in [110, 111, 112]. More recently, data-driven methods based on the fundamental lemma have been introduced in the state-space setting. For example, in [113], [114] it is used to replace the system model and initial conditions in the context of predictive control with data. In the seminal paper [95] the fundamental lemma is used to derive a data-driven representation of closed-loop systems under static state-feedback, where the controller itself is parameterised using data only. More precisely, consider an LTI system described by the dynamics

$$x(k+1) = Ax(k) + Bu(k), \tag{2.17}$$

where $x \in \mathbb{R}^n$ denotes the system state and $u \in \mathbb{R}^m$ denotes the control input. The matrices A and B of appropriate dimensions are considered *unknown.* Suppose measurements of the state response $x_{d,[k_0,k_0+T]}$ to the finite sequence of "exploring" inputs $u_{d,[k_0,k_0+T-1]}$, $k_0, T \in \mathbb{N}$, can be collected via an experiment or simulation, where the subscript d indicates measured data samples, whereas the subscript $[k_0, k_f]$ indicates the time interval over which the data is collected. The data samples can be arranged to construct the matrices

$$\begin{aligned} U_- &= \begin{bmatrix} u_d(k_0) & \dots & u_d(k_0+T-1) \end{bmatrix}, \\ X_- &= \begin{bmatrix} x_d(k_0) & \dots & x_d(k_0+T-1) \end{bmatrix}, \\ X_+ &= \begin{bmatrix} x_d(k_0+1) & \dots & x_d(k_0+T) \end{bmatrix}, \end{aligned} \tag{2.18}$$

which can be used to represent a state-feedback controller and the corresponding closed-loop system directly using data, as detailed in the following result.

Theorem 2.2.1 ([95, Theorem 2]). *Suppose the rank condition*

$$\mathit{rank}\left(\begin{bmatrix} X_- \\ U_- \end{bmatrix}\right) = n + m, \tag{2.19}$$

holds. Then, the system (2.17) *in closed-loop with* $u(k) = Kx(k)$ *can equivalently be represented as*

$$x(k+1) = X_+ G x(k), \tag{2.20}$$

with $G \in \mathbb{R}^{T\times n}$ *satisfying*

$$\begin{bmatrix} I_n \\ K \end{bmatrix} = \begin{bmatrix} X_- \\ U_- \end{bmatrix} G. \tag{2.21}$$

Using this representation, G becomes the decision variable for control design, i.e. G can be designed such that system (2.17) in closed loop with

$$u(k) = U_- G x(k), \tag{2.22}$$

satisfies certain control objectives. This result allows to formulate and solve control problems such as stabilisation and LQR in terms of convex programmes involving purely data-dependent linear matrix inequality (LMI) constraints. The resulting data-driven methods are attractive due to their comparatively low sample complexity[6] and the formulation as convex programmes, for which reliable and efficient solvers exist [115]. This has triggered renewed interest in the fundamental lemma and given rise to a large volume of work on related methods for direct data-driven control, including the *data informativity framework* [102, 116, 117]. For a recent overview of related work see the survey [118] and references therein.

Beyond "classical" discrete-time LTI systems

Various extensions of the fundamental lemma and related data-driven techniques have been proposed. In [119], the conditions of the fundamental lemma are extended to the case in which data stems from multiple data sets, rather than a single longer data trajectory. In [120], it is shown that the controllability and persistence of excitation assumptions in the context of the fundamental lemma can be relaxed. While the data-driven results discussed so far have been developed for discrete-time systems, the extension to the continuous-time setting is typically straightforward with data samples of the zero-order hold signal, if the time derivative of the system state can be measured or estimated [95]. This is further discussed in [121], which also considers the case in which data is sampled aperiodically. Continuous-time versions of the fundamental lemma have been developed in [122, 123]. The use of input-output data, rather than input-state data for data-driven control has been considered in [95, 124, 125, 126]. The data-driven representation of dynamical systems has

[6] Only data from a single input-state trajectory of length T is required. Note that a necessary condition for (2.19) to hold is $T \geq n + m$. If the input sequence $u_{d,[k_0,k_0+T-1]}$ is *persistently exciting* of order $n+1$, the rank condition (2.19) is guaranteed to hold [95]. Hence, with suitably chosen inputs, $T = (m+1)n + m$ is a sufficient trajectory length.

also been extended to linear parameter-varying systems [127, 128, 129], switched systems [130], time delay systems [131], event- and self-triggered systems [132, 133], stochastic systems [134], multi-agent systems [135] and network systems [136, 137]. Extensions for classes of nonlinear systems include bilinear systems [138, 139], polynomial systems [140] and feedback linearisable systems [141]. General nonlinear systems can be handled via Taylor's expansion [142], nonlinearity cancellation [143, 144], or by approximating the system as linear or one of the classes of nonlinear systems considered above and by treating the nonlinearities or unmodelled dynamics as disturbances (see e.g. [95, 145, 125, 141]).

Robustness to noise

Challenges arise in direct data-driven control if the system and the data are affected by disturbances and noise, which is often the case in practice. Consider the LTI dynamics

$$x(k+1) = Ax(k) + Bu(k) + d(k), \tag{2.23}$$

where $d \in \mathbb{R}^n$ denotes an unknown, additive process disturbance. In addition, it may only be possible to measure the signal

$$\zeta(k) = x(k) + v(k), \tag{2.24}$$

i.e. the state measurements $\zeta \in \mathbb{R}^n$ are corrupted by unknown measurement noise $v \in \mathbb{R}^n$. In this case, the data-driven representation of the closed-loop dynamics matrix becomes

$$A + BK = (Z_+ - W)\,G, \tag{2.25}$$

with G satisfying (2.21) with $X_- = Z_-$, and where the matrices

$$\begin{aligned} Z_- &= \begin{bmatrix} \zeta_d(k_0) & \dots & \zeta_d(k_0+T-1) \end{bmatrix}, \\ Z_+ &= \begin{bmatrix} \zeta_d(k_0+1) & \dots & \zeta_d(k_0+T) \end{bmatrix}, \end{aligned} \tag{2.26}$$

contain the noisy state measurements, and

$$W = D_- + V_+ - AV_-, \tag{2.27}$$

with the matrices

$$\begin{aligned} D_- &= \begin{bmatrix} d_d(k_0) & \dots & d_d(k_0+T-1) \end{bmatrix}, \\ V_- &= \begin{bmatrix} v_d(k_0) & \dots & v_d(k_0+T-1) \end{bmatrix}, \\ V_+ &= \begin{bmatrix} v_d(k_0+1) & \dots & v_d(k_0+T) \end{bmatrix}, \end{aligned} \tag{2.28}$$

containing the corresponding samples of the *unmeasured* process and measurement noise.

Several robust approaches have been proposed to account for noise and disturbances in the context of the direct data-driven framework discussed above. The aim is typically to design controllers guaranteeing stability or performance for all W in a specified uncertainty set, which is described via the quadratic matrix inequality

$$\begin{bmatrix} I_n \\ W^\top \end{bmatrix}^\top \begin{bmatrix} \Phi_{11} & \Phi_{12} \\ \Phi_{12}^\top & \Phi_{22} \end{bmatrix} \begin{bmatrix} I_n \\ W^\top \end{bmatrix} \succeq 0, \tag{2.29}$$

with $\Phi_{11} = \Phi_{11}^\top \in \mathbb{R}^{n\times n}$, $\Phi_{12} \in \mathbb{R}^{n\times T}$ and $\Phi_{22} = \Phi_{11}^\top \succ 0 \in \mathbb{R}^{T\times T}$. This noise model is a general condition, which does not necessarily assume any statistical properties of the noise and can capture various conditions as special cases [146, 147]. This approach has first been introduced in [95] in the context of a signal-to-noise ratio condition of the form (2.29) for robust stabilisation via a perturbed Lyapunov inequality. In [146] control design methods for robust stabilisation and performance via a linear fractional transformation and the full block S-procedure are proposed. This has been extended to incorporate prior model knowledge in [125]. Nonconservative conditions for robust stabilisation and performance are provided in [148, 149] via matrix versions of the S-lemma and Finsler's lemma, respectively, and in [150] via Petersen's lemma. Conditions based on a general theory for quadratic matrix inequalities, which encapsulates these three results, are proposed in [147]. Alternative uncertainty descriptions and conditions under which these can be converted to (2.29) are discussed in [151]. In [152], it is shown that quadratic instantaneous disturbance bounds, i.e. bounds on the disturbance at each point in time rather than the matrix W, can be less conservative than (2.29) for robust stabilisation. Alternative instantaneous bounds are used in [153] in the context of robust invariance. Most of the discussed works in the context of robust approaches to direct data-driven control focus on process noise only, i.e. the case in which $v = 0$ and hence $Z_- = X_-$, $Z_+ = X_+$ and $W = D_-$. This removes the dependence on the unknown dynamics matrix[7] A in (2.27) and hence (2.29). However, [95] considers process noise and measurement noise separately, and [150] comments on how to incorporate measurement noise. The above robust data-driven approaches based on uncertainty descriptions similar to (2.29) have also been extended to certain classes of nonlinear systems in [138, 149, 154, 140, 150, 145, 141, 155, 143].

In the context of LQR, [156] instead takes a certainty-equivalence approach to designing controllers with stability and performance guarantees directly from noisy data. It is shown that the effects of noise can be mitigated by augmenting the cost with a regulariser. Building upon this result, [157, 158, 159] demonstrate that regularisation can build a bridge between the robust and the certainty-equivalence approach, as well as between indirect and direct data-driven approaches. Simulation studies highlight

[7] Note that the appearance of A in this context can be interpreted as a measure of the "direction" of the measurement noise, which in addition to its magnitude contributes to the loss of information caused [95].

that blending the different approaches can lead to surprising performance benefits.

A behavioural approach to data-driven control based on noisy input-output data is introduced in [126]. While [126] relies on an uncertainty description of the form (2.29), a data-based behavioural uncertainty description utilising a finite-time counterpart of the gap metric is introduced in [160]. In [161, 162, 163] the authors recognise that if data is corrupted by noise, a rank condition of the form (2.19) may not be a suitable indicator of whether the data fully captures the behaviour of an LTI system. Instead, quantitative notions of persistence of excitation are introduced to formulate robust versions of the fundamental lemma.

Learning performance criteria from data

The aforementioned works address the problem of control design in the context of unknown system dynamics. In control design for optimal performance, another challenge may arise: determining suitable performance criteria. In many practical control problems, optimality criteria are not obvious *a priori* and cost functions are a design choice used to “tune” the controller. *Inverse optimal control*, first introduced in [164], is concerned with reconstructing all (if there are any) cost functions for which a given control strategy is optimal. This requires “expert information” in the form of a data trajectory corresponding to an optimal input or an optimal control law, see e.g. [165, 166]. Inverse optimal control methods have recently received increased attention, stimulated by their potential to give insight into human behaviour [167, 166, 168]. The solution of inverse optimal control problems makes it possible to understand, model and design machines which mimic or influence human or animal decision-making tasks, see e.g. [169, 170, 171]. A recent overview of inverse optimal control is given in [81].

In Chapter 4 of this thesis, the focus lies on direct data-driven control in the context of the methods discussed in this section. The framework originally introduced in [95] is extended to linear time-varying (LTV) systems, a class of systems arising in a variety of practical engineering problems [172], in Section 4.1. Methods to design state-feedback controllers with stability or trajectory boundedness and performance guarantees via purely data-dependent convex optimisation problems are proposed and discussed. Both the case in which the data is noise-free and the case in which the collected data is affected by process and measurement noise are considered. In Section 4.2, it is shown how the data-driven representation introduced in [95] can be applied not only to system dynamics, but also to quadratic objective functions in the context of optimal control. This makes it possible to formulate and solve an LQR problem with unknown system dynamics and unknown cost matrices via a convex optimisation problem purely dependent on finite-length, non-expert data of the state response to a non-optimal exploring input and a performance variable. In Chapter 5, data-driven methods are introduced in the context of dynamic games. While in Section 5.1 it is shown how the results of Section 4.2 are relevant in the context of a class of games with asymmetric information structure, the data-driven methods discussed in this section are utilised in Section 5.2 to develop data-driven Nash equilibrium-finding algorithms.

Chapter 3

Infinite-horizon dynamic games

Infinite-horizon dynamic games provide powerful tools to model dynamic interactions between strategic decision makers, as discussed in Chapter 2. However, determining equilibrium solutions for such games remains generally challenging, even in the LQ setting [9, 35, 30]. In spite of being of practical relevance in various economics and engineering applications, the class of infinite-horizon nonzero-sum LQ discrete-time dynamic games has received limited attention in the literature compared to the continuous-time case. This chapter focuses on *feedback Nash equilibrium* solutions for this class of games. Firstly, the background on infinite-horizon discrete-time dynamic games and their solution in terms of feedback Nash equilibrium strategies, which is summarised in Section 2.1, is revisited in Section 3.1 and specialised to the LQ setting. Challenges specific to the discrete-time problem are also highlighted. In Section 3.2, some intuition regarding the solutions is established by studying games involving scalar dynamics and scalar inputs via geometric approaches. Shifting the focus back to games involving general linear dynamics, a notion of approximate feedback Nash equilibrium solution is introduced and discussed in Section 3.3. Finally, iterative methods to determine feedback Nash equilibrium solutions are proposed in Section 3.4.

3.1 Extended preliminaries

Consider the infinite-horizon discrete-time dynamic game (2.5), (2.7), $i = 1, \ldots, N$. In this section, the LQ case of this class of games is considered in more detail. More precisely, let the dynamics (2.5) be of the form

$$x(k+1) = Ax(k) + \sum_{i=1}^{N} B_i u_i(k), \tag{3.1}$$

where A, B_i, for $i = 1, \ldots, N$, are constant matrices of appropriate dimension, and let the cost functional (2.7) which player i seeks to minimise via the choice of control action u_i be of the form

$$J_i(x(0), u_1(\cdot), \ldots, u_N(\cdot)) = \sum_{k=0}^{\infty} \left(x(k)^\top Q_i x(k) + \sum_{j=1}^{N} u_j(k)^\top R_{ij} u_j(k) \right), \tag{3.2}$$

where $Q_i = Q_i^\top \succeq 0$, $R_{ij} = R_{ij}^\top \succeq 0$ and $R_{ii} = R_{ii}^\top \succ 0$, for $i = 1, \ldots N$, $j = 1, \ldots, N$. Recall the definitions of the Nash equilibrium (Definition 2.1.2), and the definition of admissible strategies in the infinite-horizon context (Definition 2.1.1). In this thesis, the search of strategies for the players $i = 1, \ldots, N$ is focused on static feedback strategies, i.e. strategies of the form $\phi_i(\cdot) = \phi_i(x(k))$. An appealing property of *feedback Nash equilibrium solutions* is that the set of strategies $\{\phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))\}$ constitutes a Nash equilibrium of the game (3.1), (3.2), $i = 1, \ldots, N$, for all initial conditions $x(0) \in \mathbb{R}^n$. Moreover, if at any time instance $\bar{k} > 0$ the state $x(\bar{k})$ of system (3.1) deviates from the state $x^\star(\bar{k})$, induced by the initial condition and the equilibrium strategies $\{\phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))\}$ played for $k = 0, \ldots, \bar{k}-1$, then the set of strategies $\{\phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))\}$ still constitutes a Nash equilibrium solution of the restricted game for $k \geq \bar{k}$, a property known a *strong time consistency* [30]. Naturally, the implementation of state-feedback strategies requires that the entire state of the system is accessible to all players at all time instances. In practice, it may hence be of interest to seek an equilibrium solution in terms of strategies, which are a function of time and depend on the initial state only, i.e. $\phi_i(\cdot) = \phi_i(k, x(0))$. A comprehensive analysis of the conditions characterising such open-loop Nash equilibrium solutions of the discrete-time infinite-horizon LQ dynamic game (3.1), (3.2), $i = 1, \ldots, N$, is provided in [31].

Remark 3.1.1. A well known feature of deterministic optimal control problems (single player dynamic games) is that the open-loop and the feedback solution are equivalent in the sense that they result in the same optimal cost and the same time trajectory of the controlled system, because the trajectories of the system are completely determined by the initial conditions in the absence of uncertainties [82]. It is important to highlight that this is not the case in the dynamic game setting if $N > 1$, i.e. even though open-loop Nash equilibrium solutions of LQ dynamic games sometimes admit a feedback implementation, they are in general not feedback Nash equilibrium solutions (see e.g. [31]). This is because the evolution of system (3.1) is not completely determined by the initial conditions and the action of player i, but it is also influenced by the actions of players $j = 1, \ldots, N$, $j \neq i$. Considering the problem faced by player i, i.e. the minimisation of the cost functional (3.2) subject to the dynamics (3.1), the solution hence depends on the assumptions regarding the types of strategies played by the other players $j = 1, \ldots, N$, $j \neq i$. If the strategies of players j are fixed (either at open-loop or feedback information structure), the open-loop solution and the feedback solution of the problem faced by player i are equivalent. However, the search for an open-loop solution for player i assuming players j play open-loop strategies is not equivalent to the search for a feedback solution for player i assuming players j also play feedback strategies. In other words, a distinction between open-loop and feedback Nash equilibrium solutions is the set of alternative strategies for all N players to which a candidate Nash equilibrium solution is compared in (2.8) (see Definition 2.1.2).

Considering feedback strategies, it can be shown via the dynamic programming principle [50] that Nash equilibria are characterised by the stabilising solutions of a set of coupled difference equations, as recalled in Chapter 2. In particular, consider linear static state-feedback strategies of the form $\phi_i(\cdot) = K_i x(k)$, as common in the LQ setting (see e.g. [30, 55, 57, 56, 62, 59, 31])[1], and consider the following assumption, which is a necessary and sufficient condition for the set of admissible strategies in the sense of Definition 2.1.1 to be non-empty.

Assumption 3.1.1. The pair $\left(A, \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix}\right)$ is stabilisable.

In this setting, feedback Nash equilibrium solutions are characterised by the stabilising solutions of a set of coupled algebraic equations, as detailed in the following result. Similar conditions can also be found in [9, Section 6.2.3] and [173, Section 6.7.2], without a detailed proof. While the proof is analogous to the continuous-time counterpart (see e.g. [30, Section 8.3]), it is included here for completeness, since the discrete-time version is not as readily available in the literature as the continuous-time version.

Theorem 3.1.1. *Consider the game* (3.1), (3.2), $i = 1, \dots, N$. *The set of strategies*

$$\{\phi_1^\star(x(k)), \dots, \phi_N^\star(x(k))\},$$

where

$$\phi_i^\star(x(k)) = K_i^\star x(k), \tag{3.3}$$

for $i = 1, \dots, N$, *constitutes a feedback Nash equilibrium solution of the game if and only if*

$$\rho\left(A + \sum_{j=1}^{N} B_j K_j^\star\right) < 1, \tag{3.4}$$

and there exist $P_i^\star = P_i^{\star\top} \succeq 0 \in \mathbb{R}^{n\times n}$ *satisfying*

$$P_i^\star = Q_i + \sum_{j=1}^{N} K_j^{\star\top} R_{ij} K_j^\star + \left(A + \sum_{j=1}^{N} B_j K_j^\star\right)^\top P_i^\star \left(A + \sum_{j=1}^{N} B_j K_j^\star\right), \tag{3.5}$$

[1]Note, however, that LQ dynamic games may admit nonlinear feedback Nash equilibrium solutions [53].

for $i = 1, \ldots, N$, and

$$\begin{bmatrix} R_{11} + B_1^\top P_1^\star B_1 & \ldots & B_1^\top P_1^\star B_N \\ \vdots & \ddots & \vdots \\ B_N^\top P_N^\star B_1 & \ldots & R_{NN} + B_N^\top P_N^\star B_N \end{bmatrix} \begin{bmatrix} K_1^\star \\ \vdots \\ K_N^\star \end{bmatrix} = - \begin{bmatrix} B_1^\top P_1^\star \\ \vdots \\ B_N^\top P_N^\star \end{bmatrix} A. \tag{3.6}$$

The feedback Nash equilibrium is such that the equilibrium cost incurred by player i starting from initial condition $x(0)$ is $J_i^\star = J_i(x(0), \phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))) = x(0)^\top P_i^\star x(0)$.

Proof. Consider first the sufficient implication and let the set of static state-feedback gains $\{\bar{K}_1^\star, \ldots, \bar{K}_N^\star\}$ and the set of symmetric positive semi-definite matrices $\{\bar{P}_1^\star, \ldots, \bar{P}_N^\star\}$ satisfy (3.4), (3.5), $i = 1, \ldots, N$, and (3.6). Assuming that the actions of players j, $j = 2, \ldots, N$, are fixed at $u_j(k) = \bar{K}_j^\star x(k)$, consider the minimisation of the cost function (3.2), $i = 1$, subject to the system dynamics (3.1) by player 1, namely the minimisation of

$$J_1(x(0), u_1(\cdot), \bar{K}_2^\star x(k), \ldots, \bar{K}_N^\star x(k)) = \sum_{k=0}^{\infty} \left(x(k)^\top \underbrace{\left(Q_1 + \sum_{j=2}^{N} \bar{K}_j^{\star\top} R_{1j} \bar{K}_j^\star \right)}_{\bar{Q}_1} x(k) + u_1(k)^\top R_{11} u_1(k) \right), \tag{3.7}$$

subject to

$$x(k+1) = \underbrace{\left(A + \sum_{j=2}^{N} B_j \bar{K}_j^\star \right)}_{\bar{A}_1} x(k) + B_1 u_1(k). \tag{3.8}$$

This constitutes an LQR problem for player 1. Note that by assumption, there exists a stabilising solution $P^{OC} = \bar{P}_1^\star$ satisfying the algebraic Riccati equation

$$P^{OC} = \bar{Q}_1 + \bar{A}_1^\top P^{OC} \bar{A}_1 - \bar{A}_1^\top P^{OC} B_1 \left(R_{11} + B_1^\top P^{OC} B_1 \right)^{-1} B_1^\top P^{OC} \bar{A}_1, \tag{3.9}$$

associated with this LQR problem. This can be seen by rearranging the first line of (3.6) and inserting it into (3.5), $i = 1$. With the strategies of players j, $j = 2, \ldots, N$, fixed, the optimal strategy for player 1 is hence

$$u_1^{OC}(k) = - \left(R_{11} + B_1^\top P^{OC} B_1 \right)^{-1} B_1^\top P^{OC} \bar{A}_1 x(k) = \bar{K}_1^\star x(k), \tag{3.10}$$

and the optimal cost is $J_1(x(0), \bar{K}_1^\star x(k), \ldots, \bar{K}_N^\star x(k)) = x(0)^\top P^{OC} x(0) = x(0)^\top \bar{P}_1^\star x(0)$, see e.g. [52]. Hence,

$$J_1(x(0), \bar{K}_1^\star x(k), \ldots, \bar{K}_N^\star x(k)) \leq J_1(x(0), u_1(\cdot), \bar{K}_2^\star x(k), \ldots, \bar{K}_N^\star x(k)), \tag{3.11}$$

for all admissible $\{u_1(\cdot), \bar{K}_2^\star x(k), \ldots, \bar{K}_N^\star x(k)\}$. Conversely, assume the state-feedback gains $\{\bar{K}_1^\star, \ldots, \bar{K}_N^\star\}$ correspond to a feedback Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \ldots, N$. By Definition 2.1.2, (3.11) holds for all admissible $\{u_1(\cdot), \bar{K}_2^\star x(k), \ldots, \bar{K}_N^\star x(k)\}$. Hence, with the actions of players j, $j = 2, \ldots, N$, fixed at $u_j(k) = \bar{K}_j^\star x(k)$, (3.10) with P^{OC} the solution of (3.9) is the unique stabilising optimal control action for player 1 solving the LQR problem (3.8), (3.7), see e.g. [52]. This implies that there exists $\bar{P}_1^\star = P^{OC}$ such that (3.5), $i = 1$, and the first line of (3.6) hold. The proof is concluded via analogous arguments for players j, $j = 2, \ldots, N$. □

In Theorem 3.1.1 feedback Nash equilibrium solutions of LQ dynamic games are characterised via the stabilising solutions of a set of coupled algebraic equations, namely (3.5), $i = 1, \ldots, N$, and (3.6), which may also admit solutions which do not render the closed-loop system stable. Sufficient conditions for solutions of (3.5), $i = 1, \ldots, N$, (3.6) to be stabilising are provided in the following result. Note that the conditions are easily verifiable based on the system and cost matrices and more general than those in [9, Proposition 6.3].

Corollary 3.1.1. *Let $\{K_1^\star, \ldots, K_N^\star\}$ and $\{P_1^\star, \ldots, P_N^\star\}$, where $P_i^\star = P_i^{\star\top} \succeq 0$, for $i = 1, \ldots, N$, be a solution to (3.5), $i = 1, \ldots, N$, and (3.6). If the pair $\left(A, \sum_{i=1}^N Q_i\right)$ is detectable[2], then $\{K_1^\star, \ldots, K_N^\star\}$ is such that (3.4) holds and the corresponding strategies (3.3), $i = 1, \ldots, N$, constitute a feedback Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \ldots, N$.*

Proof. The claim is demonstrated by contradiction. Consider the sum over i of (3.5), for $i = 1, \ldots, N$, namely

$$\sum_{i=1}^N P_i^\star = \sum_{i=1}^N Q_i + \sum_{j=1}^N K_j^{\star\top} \left(\sum_{i=1}^N R_{ij}\right) K_j^\star + \left(A + \sum_{j=1}^N B_j K_j^\star\right)^\top \left(\sum_{i=1}^N P_i^\star\right) \left(A + \sum_{j=1}^N B_j K_j^\star\right). \quad (3.12)$$

Assume there exists an eigenvalue λ and corresponding eigenvector $v \neq 0 \in \mathbb{R}^n$ of $A_{cl}^\star = \left(A + \sum_{j=1}^N B_j K_j^\star\right)$, such that $|\lambda| \geq 1$, i.e. $\rho\left(A_{cl}^\star\right) \geq 1$. Recall that by the definition of eigenvalues and eigenvectors $A_{cl}^\star v = \lambda v$. Hence, pre- and post-multiplying (3.12) by v^{H} and v, respectively, gives

$$v^{\mathsf{H}} \left(\lambda^{\mathsf{H}}\lambda - 1\right) \left(\sum_{i=1}^N P_i^\star\right) v + v^{\mathsf{H}} \left(\sum_{i=1}^N Q_i\right) v + \sum_{j=1}^N v^{\mathsf{H}} K_j^{\star\top} \left(\sum_{i=1}^N R_{ij}\right) K_j^\star v = 0. \quad (3.13)$$

Since by assumption $\lambda^{\mathsf{H}}\lambda = |\lambda|^2 \geq 1$ and by definition $\sum_{i=1}^N P_i^\star \succeq 0$, $\sum_{i=1}^N Q_i \succeq 0$, and $\sum_{i=1}^N R_{ij} \succ 0$, (3.13) implies that $v^{\mathsf{H}}\left(\lambda^{\mathsf{H}}\lambda - 1\right)\left(\sum_{i=1}^N P_i^\star\right) v = 0$, $v^{\mathsf{H}}\left(\sum_{i=1}^N Q_i\right) v = 0$ and $\sum_{j=1}^N v^{\mathsf{H}} K_j^{\star\top}\left(\sum_{i=1}^N R_{ij}\right) K_j^\star v = 0$. This in turn implies that either $\left(\lambda^{\mathsf{H}}\lambda - 1\right) = 0$ or $\left(\sum_{i=1}^N P_i^\star\right) v = 0$, and that $\left(\sum_{i=1}^N Q_i\right) v = 0$ and $K_j^\star v = 0$, for $j = 1, \ldots, N$. The latter gives $A_{cl}^\star v = Av = \lambda v$, i.e. λ and v are also an eigenvalue and an eigenvector of A. Note that $\left(\sum_{i=1}^N Q_i\right) v = 0$ implies that the

[2]Note that this is always the case if $\sum_{i=1}^N Q_i \succ 0$.

corresponding unstable mode is not observable through $\sum_{i=1}^N Q_i$. This contradicts the hypothesis that the pair $\left(A, \sum_{i=1}^N Q_i\right)$ is detectable. □

Remark 3.1.2. The equations (3.5), $i = 1, \ldots, N$, and (3.6) are reminiscent of the classic *algebraic Riccati equation* arising in the discrete-time LQR problem (single-player LQ discrete-time dynamic game), with additional terms accounting for the presence of the other players. Note, however, that in contrast to the quadratic algebraic Riccati equations arising in the discrete-time LQR problem, (3.5), $i = 1, \ldots, N$, and (3.6) are *not quadratic in the decision variables*, even after eliminating one set of variables (either $K_i^\star$ or $P_i^\star$, for $i = 1, \ldots, N$). It is also interesting to put this observation into perspective with respect to the continuous-time counterpart. Nash equilibrium solutions to an N-player nonzero-sum LQ differential game are characterised by the stabilising solutions of N *coupled algebraic Riccati equations* (see e.g. [30, Chapter 8]), which in contrast to (3.5), $i = 1, \ldots, N$, and (3.6) are quadratic equations. The solutions of these equations $P_i^{c\star}$, $i = 1, \ldots, N$, are related to the value functions associated to each player i and, notably, the feedback gain $K_i^{c\star}$ of player i, $i = 1, \ldots, N$, corresponding to the feedback Nash equilibrium strategy $\phi_i^\star(x(t)) = K_i^{c\star} x(t)$, depends explicitly only on the matrix $P_i^{c\star}$. The dependency on the matrices $P_j^{c\star}$, $j = 1, \ldots, N$, $j \neq i$, associated with the other players is instead *only implicit* through the coupling of the N equations. In the discrete-time case, on the other hand, it is evident from (3.6) that the feedback Nash equilibrium gain $K_i^\star$ *explicitly depends* on all $P_j^\star$, for $j = 1, \ldots, N$. These differences make the computation of solutions to (3.5), $i = 1, \ldots, N$, and (3.6), even more challenging compared to the continuous-time setting, and make the discrete-time case interesting to study.

3.2 Feedback Nash equilibria in scalar games

The *Nash equilibrium* is a commonly considered solution concept in dynamic games. It is of natural interest in non-cooperative settings since the equilibrium is "secure" against a unilateral deviation by any player, i.e. such a deviation always results in a worse outcome for the deviating player [9]. For infinite-horizon LQ dynamic games, feedback Nash equilibria, i.e. Nash equilibrium solutions involving linear static state-feedback strategies, are characterised via the solutions of coupled algebraic matrix equations. These coupled equations look reminiscent of the algebraic Riccati equations arising in LQ optimal control. However, in contrast to the Riccati equations, which under mild assumptions admit a unique solution and for which efficient solution methods exist, the coupled matrix equations associated with Nash equilibrium solutions of LQ dynamic games are generally difficult to solve, as noted in [9, 35, 30] and highlighted for the discrete-time case in Section 3.1, and may admit multiple solutions with different outcomes. While the existence, number and proprieties of feedback Nash equilibrium solutions for LQ dynamic games have been extensively studied in the continuous-time setting (i.e. for LQ differential games), for example in [9, 54, 55, 56, 57, 58, 30], the discrete-time case has received less attention [9, 59]. As highlighted in Section 3.1, additional product terms of the decision variables arising in the coupled matrix equations in the discrete-time setting lead to additional challenges and make the problem interesting to consider.

To build intuition regarding feedback Nash equilibrium solutions of discrete-time LQ dynamic games and their properties, *scalar games* are studied in this section. More precisely, the focus lies on games involving dynamics in which the state and inputs are scalar variables, and the cost functional associated with each player only penalises the player's own control action. In this case, the coupled algebraic equations associated with feedback Nash equilibria simplify from a set of matrix equations to a set of scalar equations. This allows to use geometric arguments to specify conditions in terms of the system and cost parameters under which a game admits a certain number of feedback Nash equilibrium solutions. Such information is not only desirable from a computational point of view, but also provides a better understanding of the different possible feedback Nash equilibrium outcomes of a given game. Of particular interest in the analysis of scalar games is the two-player case. For this special case, analytical results can be derived. In addition, the coupled equations associated with feedback Nash equilibria, and their solutions, can be represented and analysed graphically, which allows to gain further insights. The analysis presented in this section can be understood as a discrete-time counterpart to the study of scalar differential games presented in [58] and [30, Chapter 8.4].

The remainder of this section is organised as follows. The considered problem is defined in Section 3.2.1. A graphical interpretation of the coupled equations associated with feedback Nash equilibrium solutions is introduced in Section 3.2.2 and utilised in Section 3.2.3 to characterise conditions for the number of feedback Nash equilibria in terms of the system and cost parameters. Additional insights for the two-player case are provided in Section 3.2.4. Finally, the results are illustrated via numerical examples in Section 3.2.5.

3.2.1 Problem formulation

Consider the LQ infinite-horizon discrete-time dynamic game (3.1), (3.2), $i = 1, \ldots, N$, in which $n = 1$, $m_i = 1$, for $i = 1, \ldots, N$, and $R_{ij} = 0$, for $j = 1, \ldots, N$, $i = 1, \ldots, N$, $j \neq i$. The latter implies that the cost functional (3.2) associated with player i only penalises the control effort u_i of player i and not the actions u_j of the other players j, $j = 1, \ldots, N$, $j \neq i$. This assumption is crucial for the derivations in the following sections. While it results in some loss of generality, the class of LQ games in which $R_{ij} = 0$ is commonly considered in the literature, see e.g. [58, 30, 62, 29, 31]. To emphasise that scalar quantities are considered, lower case letters are used for the system and cost parameters throughout this section. Namely, consider the game defined by the dynamics

$$x(k+1) = ax(k) + \sum_{i=1}^{N} b_i u_i(k), \tag{3.14}$$

where $a \in \mathbb{R}$, $b_i \in \mathbb{R} \setminus \{0\}$, $i = 1, \ldots, N$, are constant system parameters[3], and the cost functionals

$$J_i(x(0), u_1(\cdot), \ldots, u_N(\cdot)) = \sum_{k=0}^{\infty} \left(q_i x(k)^2 + r_i u_i(k)^2 \right), \tag{3.15}$$

with $q_i \in \mathbb{R}$ and $r_i > 0 \in \mathbb{R}$, for $i = 1, \ldots, N$. Note that in contrast to the other sections of this chapter, in this section (apart from Subsection 3.2.4) q_i is not restricted to be non-negative. This is in line with the game definition considered for the analysis of scalar dynamic games in the continuous-time setting in [58] and [30, Chapter 8.4].

Consider the problem of determining feedback Nash equilibrium solutions, i.e. feedback strategies of the form $\phi_i(x(k)) = k_i x(k)$ for the players' actions u_i, $i = 1, \ldots, N$, which render the zero equilibrium of the system (3.14) asymptotically stable and are such that (2.8) holds (see Definition 2.1.2). For LQ games with general linear dynamics (3.1), (3.2), $i = 1, \ldots, N$, it is shown in Theorem 3.1.1 that feedback Nash equilibria are characterised by the stabilising solutions of the coupled algebraic equations (3.5), $i = 1, \ldots, N$, and (3.6). In contrast to the setting considered in Theorem 3.1.1 (and commonly considered in single player LQR problems), in this section q_i in (3.15) may be negative. In the following, the result of Theorem 3.1.1 is revisited and revised for the scalar game problem considered in this section.

[3]If $b_i = 0$, for any $i = 1, \ldots, N$, then u_i does not influence the dynamics (3.14) and player i can be disregarded. Hence, this case is excluded without loss of generality.

Corollary 3.2.1. *Consider the game* (3.14)*,* (3.15)*,* $i = 1, \ldots, N$*. The set of strategies*

$$\{\phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))\},$$

where

$$\phi_i^\star(x(k)) = k_i^\star x(k), \tag{3.16}$$

for $i = 1, \ldots, N$*, constitutes a feedback Nash equilibrium solution of the game if and only if*

$$\left| a + \sum_{i=1}^{N} b_i k_i^\star \right| < 1, \tag{3.17}$$

and there exist $p_i \in \mathbb{R}$*, for* $i = 1, \ldots, N$*, satisfying the set of equations*

$$0 = \left(a + \sum_{j=1}^{N} b_j k_j^\star \right)^2 p_i^\star - p_i^\star + q_i + k_i^{\star 2} r_i, \tag{3.18a}$$

$$0 = \left(r_i + b_i^2 p_i^\star \right) k_i^\star + b_i p_i^\star \left(a + \sum_{j=1, j\neq i}^{N} b_j k_j^\star \right), \tag{3.18b}$$

such that

$$\left(r_i + b_i^2 p_i^\star \right) > 0, \tag{3.19}$$

for $i = 1, \ldots, N$*. The feedback Nash equilibrium is such that the equilibrium cost incurred by player* i *starting from initial condition* $x(0)$ *is* $J_i^\star = J_i(x(0), \phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))) = p_i^\star x(0)^2$*.*

Proof. The result follows from analogous arguments to the proof of Theorem 3.1.1. More precisely, let $\{\bar{k}_1^\star, \ldots, \bar{k}_N^\star\}$ and $\{\bar{p}_1^\star, \ldots, \bar{p}_N^\star\}$ satisfy (3.17), (3.18) and (3.19), for $i = 1, \ldots, N$. Assuming the actions of players j, $j = 2, \ldots, N$, are fixed at $u_j(k) = \bar{k}_j^\star x(k)$, consider the minimisation of the cost function (3.15), $i = 1$, subject to the dynamics (3.14) by player 1, namely the minimisation of

$$J_1(x(0), u_1(\cdot), \bar{k}_2^\star x(k), \ldots, \bar{k}_N^\star x(k)) = \sum_{k=0}^{\infty} \left(q_1 x(k)^2 + r_1 u_1(k)^2 \right), \tag{3.20}$$

subject to

$$x(k+1) = \underbrace{\left(a + \sum_{j=2}^{N} b_j \bar{k}_j^\star \right)}_{\bar{a}_1} x(k) + b_1 u_1(k). \tag{3.21}$$

This constitutes an LQR problem for player 1. Via the dynamic programming principle [50] and Pontryagin's Maximum principle [51] it follows without any assumptions regarding the sign of q_i that the LQR problem (3.21), (3.20) admits a stabilising state-feedback solution $u_1^{OC}(k) = k^{OC} x(k)$ if and only if $\left|\bar{a}_1 + b_1 k^{OC}\right| < 1$ and

$$0 = \left(\bar{a}_1 + b_1 k^{OC}\right)^2 p^{OC} - p^{OC} + q_1 + k^{OC^2} r_1, \tag{3.22a}$$

$$0 = \left(r_1 + b_1^2 p^{OC}\right) k^{OC} + b_1 p^{OC} \bar{a}_1\,, \tag{3.22b}$$

hold for some p^{OC}, such that $\left(r_1 + b_1^2 p^{OC}\right) > 0$. Note that the latter condition ensures that the minimisation problem is well-posed. The corresponding minimum cost is given by $J_1(x(0), k^{OC}x(k), \bar{k}_2^\star x(k), \ldots, \bar{k}_N^\star x(k)) = p^{OC} x(0)^2$. Note that by assumption and by comparing (3.18), $i = 1$, and (3.22), there exists a stabilising solution $k^{OC} = \bar{k}_1^\star$ and $p^{OC} = \bar{p}_1^\star$ to (3.22). Hence,

$$J_1(x(0), \bar{k}_1^\star x(k), \ldots, \bar{k}_N^\star x(k)) \leq J_1(x(0), u_1(\cdot), \bar{k}_2^\star x(k), \ldots, \bar{k}_N^\star x(k)), \tag{3.23}$$

for all admissible $\{u_1(\cdot), \bar{k}_2^\star x(k), \ldots, \bar{k}_N^\star x(k)\}$. Conversely, assume the stabilising set of state-feedback gains $\{\bar{k}_1^\star, \ldots, \bar{k}_N^\star\}$ corresponds to a feedback Nash equilibrium solution of the game (3.14), (3.15), $i = 1, \ldots, N$. By Definition 2.1.2, (3.23) holds for all admissible $\left\{u_1(\cdot), \bar{k}_2^\star x(k), \ldots, \bar{k}_N^\star x(k)\right\}$ if the actions of players j, $j = 2, \ldots, N$, are fixed at $u_j(k) = \bar{k}_j^\star x(k)$. Hence, there exists p^{OC} satisfying (3.22) for $k^{OC} = \bar{k}_1^\star$, which by comparing (3.22) and (3.18), $i = 1$, implies that $\bar{k}_1^\star$ and $\bar{p}_1^\star = p^{OC}$ satisfy (3.17), (3.18) and (3.19), for $i = 1$. The proof is concluded via analogous arguments for players j, $j = 2, \ldots, N$. □

In Corollary 3.2.1, feedback Nash equilibrium solutions of the scalar game (3.14), (3.15), $i = 1, \ldots, N$, are characterised by the solutions of the coupled algebraic equations (3.18a), (3.18b), $i = 1, \ldots, N$, which are such that (3.17) and (3.19) hold[4]. Even though these conditions constitute a set of coupled scalar equations, it is still challenging to determine a solution, especially if the game involves a large amount of players N. Moreover, multiple solutions to (3.17), (3.18) and (3.19), $i = 1, \ldots, N$, and hence multiple feedback Nash equilibria of the game (3.14), (3.15), $i = 1, \ldots, N$, may exist. The problem considered in this section is hence to characterise all feedback Nash equilibria of the game (3.14), (3.15), $i = 1, \ldots, N$, via a geometric approach, and to utilise this to derive conditions under which the game admits a certain number of feedback Nash equilibria.

3.2.2 Graphical interpretation of the coupled algebraic equations

With the aim of deriving conditions for the existence of no, a unique or multiple feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, \ldots, N$, consider the following assumptions.

[4]While in the continuous-time setting the condition $r_i > 0$ is sufficient to ensure the solutions of the equivalent of (3.18a), (3.18b) correspond to minimising the equivalent of (3.15) for player i, the additional condition (3.19) is needed in the discrete-time case, for $i = 1, \ldots, N$. This is another interesting difference, which in addition to those highlighted in Remark 3.1.2 make the discrete-time setting considered herein interesting to study.

Assumption 3.2.1. Let
$$\sigma_i = \frac{b_i^2 q_i}{r_i},$$
for $i = 1, \ldots, N$. The players are ordered such that $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_N$.

Assumption 3.2.2. Let $\sigma_N > -1$.

Lemma 3.2.1. *Consider the game* (3.14)*,* (3.15)*, $i = 1, \ldots, N$. If Assumptions 3.2.1 and 3.2.2 hold, then* (3.19) *holds for any solution of* (3.17)*,* (3.18)*, for $i = 1, \ldots, N$.*

Proof. The condition (3.19) holds for a solution $p_i^\star$, $k_i^\star$ of (3.17), (3.18), if $p_i^\star > -\frac{r_i}{b_i^2}$, for $i = 1, \ldots, N$. Consider first the case $a_{cl}^\star := a + \sum_{j=1}^N b_j k_j^\star \neq 0$. Combining (3.18a) and (3.18b), and solving (3.18a) for $p_i^\star$ gives that $p_i^\star > -\frac{r_i}{b_i^2}$ is equivalent to
$$\frac{b_i k_i^\star}{a_{cl}^\star} < 1, \tag{3.24}$$
for $i = 1, \ldots, N$. Combining again (3.18a) and (3.18b) and solving for $k_i^\star$, (3.24) is in turn equivalent to
$$1 - \frac{a_i}{\gamma_i \pm \sqrt{\gamma_i^2 - 1}} < 1,$$
with $a_i = a + \sum_{j=1, j\neq i}^N b_j k_j^\star$ and $\gamma_i = \frac{1}{2}\left(a_i + \frac{\sigma_i + 1}{a_i}\right)$, which holds if a_i and γ_i have the same sign. This in turn holds true if $\sigma_i > -1$. If $a_{cl}^\star = 0$, (3.18a) and (3.18b) imply $p_i^\star = q_i$. Hence, (3.19) holds if $q_i > -\frac{r_i}{b_i^2}$, which is again equivalent to $\sigma_i > -1$. By Assumption 3.2.1, Assumption 3.2.2 implies $\sigma_i > -1$, for $i = 1, \ldots, N$. □

Remark 3.2.1. While Assumption 3.2.1 can be introduced without loss of generality, Assumption 3.2.2 only depends on system and cost parameters and can hence be verified *prior* to the computation of solutions. The result of Lemma 3.2.1 above highlights the relevance of Assumption 3.2.2. Note that the following results regarding the characterisation, number and properties of solutions are still relevant if Assumption 3.2.2 does not hold. However, in this case the results concern any solutions of (3.17), (3.18), $i = 1, \ldots, N$. Hence, the condition (3.19), or alternatively (3.24), needs to be checked to ensure such a solution corresponds to a feedback Nash equilibrium.

The following result constitutes a reformulation of Corollary 3.2.1.

Lemma 3.2.2. *Consider the game* (3.14)*,* (3.15)*, $i = 1, \ldots, N$. Let Assumptions 3.2.1 and 3.2.2 hold and consider the function*
$$\hat{f}(\xi) = \begin{cases} -\xi - \sqrt{\xi^2 + 1} & \textit{if } \xi < 0, \\ -\xi + \sqrt{\xi^2 + 1} & \textit{if } \xi > 0. \end{cases} \tag{3.25}$$

The set of strategies $\{\phi_1^\star(x(k)),\ldots,\phi_N^\star(x(k))\}$*, with* $\phi_i^\star(x(k))$ *given by* (3.16)*, with*

$$k_i^\star = \frac{-\xi^\star - t_i\sqrt{\xi^{\star 2} - \sigma_i}}{b_i}, \tag{3.26}$$

for $i = 1,\ldots,N$*, constitutes a feedback Nash equilibrium of the game with* $a_{cl}^\star = a + \sum_{j=1}^N b_j k_j^\star \neq 0$*, if and only if there exist* $t_i \in \{-1,1\}$*, for* $i = 1,\ldots,N$*, and* $\xi^\star \neq 0$*, satisfying*

$$a = \hat{f}(\xi^\star) + N\xi^\star + t_1\sqrt{\xi^{\star 2} - \sigma_1} + \ldots + t_N\sqrt{\xi^{\star 2} - \sigma_N}. \tag{3.27}$$

Proof. By Corollary 3.2.1, feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1,\ldots,N$, are characterised by the stabilising solutions of (3.18), (3.19), $i = 1,\ldots,N$. The proof lies in showing that solving (3.27) is equivalent to solving (3.17), (3.18) and (3.19), $i = 1,\ldots,N$. Eliminating $p_i^\star$ in (3.18) (by solving (3.18a) for $p_i^\star$ and substituting this into (3.18b)) gives the condition

$$0 = \frac{b_i}{2}k_i^{\star 2} + \xi^\star k_i^\star + \frac{\sigma_i}{2b_i}, \tag{3.28}$$

with $\xi^\star := \frac{1}{2}\left(\frac{1}{a_{cl}^\star} - a_{cl}^\star\right)$. The equation (3.28) admits the solutions (3.26), $t_i \in \{-1,1\}$, for $i = 1,\ldots,N$. Hence, $a_{cl}^\star$ can be written as

$$\begin{aligned} a_{cl}^\star = -\xi^\star \pm \sqrt{\xi^{\star 2}+1} &= a + \sum_{j=1}^N b_j k_j^\star, \\ &= a - N\xi^\star - t_1\sqrt{\xi^{\star 2} - \sigma_1} - \ldots - t_N\sqrt{\xi^{\star 2} - \sigma_N}. \end{aligned} \tag{3.29}$$

By (3.17), the solutions (3.26), $t_i \in \{-1,1\}$ of interest are such that $|a_{cl}^\star| < 1$. Hence, there is a one-to-one correspondence between $\xi^\star$ and $a_{cl}^\star$ given by $a_{cl}^\star = \hat{f}(\xi^\star)$, with $\hat{f}(\xi)$ as defined in (3.25) and illustrated in Figure 3.1. Substituting this into (3.29) gives (3.27). Thus, any solution to (3.17), (3.18) and (3.19), $i = 1,\ldots,N$, is such that (3.27) holds. Conversely, let $\{t_1,\ldots,t_N\}$ and $\xi^\star$ be a

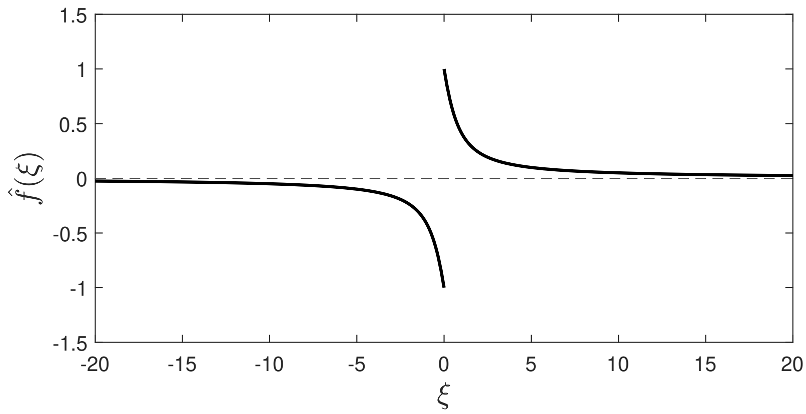


Figure 3.1: Plot of function $\hat{f}(\xi)$ defined in (3.25).

solution of (3.27) and note that this is such that (3.28) holds with $k_i^\star$ as in (3.26), for $i = 1, \dots, N$. Via $a_{cl}^\star = \hat{f}(\xi^\star)$ as defined in (3.25), this implies that (3.17) holds and (3.18b) holds with $p_i^\star = \frac{q_i + k_i^{\star 2} r_i}{1 - a_{cl}^{\star 2}}$, $i = 1, \dots, N$, which is the unique solution of the Lyapunov equation (3.18a) for fixed $a_{cl}^\star$. By Lemma 3.2.1, Assumption 3.2.2 ensures that (3.19) holds, for $i = 1, \dots, N$. Hence, (3.27) implies (3.17), (3.18) and (3.19), $i = 1, \dots, N$. □

Remark 3.2.2. The result of Lemma 3.2.2 relies on the assumption $a_{cl}^\star \neq 0$. Note that if $a_{cl}^\star = 0$, (3.18b) implies $k_i^\star = 0$, for $i = 1, \dots, N$, and hence $a = 0$. The assumption $a_{cl}^\star \neq 0$ is thus only restrictive in the special case in which $a = 0$. In this case, a set of feedback Nash equilibrium strategies is given by (3.16), with $k_i^\star = 0$, $i = 1, \dots, N$. However, this trivial solution cannot be found via Lemma 3.2.2. All feedback Nash equilibria of the game (3.14), (3.15), $i = 1, \dots, N$, with $a = 0$ are hence given by the solutions satisfying the conditions of Lemma 3.2.2 (if there are any) plus the solution (3.16), with $k_i^\star = 0$, $i = 1, \dots, N$.

Lemma 3.2.2 introduces an alternative characterisation of feedback Nash equilibria of the game (3.14), (3.15), $i = 1, \dots, N$, via the condition (3.27). Consider the auxiliary functions

$$f_\ell(\xi) = \hat{f}(\xi) + N\xi + \tau_{\ell,1}\sqrt{\xi^2 - \sigma_1} + \ldots + \tau_{\ell,N}\sqrt{\xi^2 - \sigma_N}, \tag{3.30}$$

for $\ell = 1, \dots, \mathcal{L}$, where $\mathcal{L} = 2^N$ and $\tau_\ell = (\tau_{\ell,1}, \dots, \tau_{\ell,N})$ is an N-tuple over the set $\{-1, 1\}$. Hence, the functions $f_\ell(\xi)$, $\ell = 1, \dots, \mathcal{L}$, in (3.30) capture all possible combinations of the values which t_i, for $i = 1, \dots, N$, can take in (3.27). By Lemma 3.2.2, feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, \dots, N$, are then represented graphically by the intersections of the functions $f_\ell(\xi)$, $\ell = 1, \dots, \mathcal{L}$, as defined in (3.30), with the level a.

Remark 3.2.3. If the level a intersects two auxiliary functions $f_\ell(\xi)$ and $f_w(\xi)$ as defined in (3.30) at a point $(\bar{\xi}, \bar{f})$ in which they coincide, i.e. $\bar{f} = f_\ell(\bar{\xi}) = f_w(\bar{\xi})$, for $\ell = 1, \dots, \mathcal{L}$, $w = 1, \dots, \mathcal{L}$, $\ell \neq w$, then this intersection point generally corresponds to two distinct feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, \dots, N$, resulting in the same closed-loop dynamics $a_{cl}^\star$. However, there exists only one corresponding set of gains $\{k_1^\star, \dots, k_N^\star\}$ for an intersection of level a with $f_\ell(\xi)$ and $f_w(\xi)$ in the point $(\bar{\xi}, \bar{f})$, if $\bar{\xi} = \pm\sqrt{\sigma_j}$, $j = 1, \dots, N$, and if $f_\ell(\xi)$ and $f_w(\xi)$ are such that

$$\tau_{\ell,i} = \tau_{w,i},$$

for all $i = 1, \dots, N$, $i \neq j$, or for all $i \neq l$, $i \neq j$ if the game is such that any

$$\sigma_l = \sigma_j,$$

$l = 1, \dots, N$, $l \neq j$. The corresponding set of gains is given by $k_j^\star = -\frac{\bar{\xi}}{b_j}$, $k_l^\star = -\frac{\bar{\xi}}{b_l}$ and $k_i^\star$ as defined in (3.26) with $t_i = \tau_{\ell,i} = \tau_{w,i}$, for $i \neq j$, $i \neq l$. Hence, in this special case the intersection point $(\bar{\xi}, \bar{f})$ of the level a with the two functions $f_\ell(\xi)$ and $f_w(\xi)$ corresponds to one feedback Nash equilibrium solution rather than two distinct feedback Nash equilibrium solutions. These observations readily extend to the case in which the line at level a intersects more than two auxiliary functions in a single point.

Certain properties of the auxiliary functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, are highlighted in the following result. To this end, let $\mathcal{T} = \{\tau_1, \ldots, \tau_L\}$ denote the set of all N-tuples τ_ℓ, $\ell = 1, \ldots, \mathcal{L}$, and consider the function $T_\ell : \{1, \ldots, N\} \to \{\tau_{\ell,1}, \ldots, \tau_{\ell,N}\}$, defined by $T_\ell(i) = \tau_{\ell,i}$, for $i = 1, \ldots, N$.

Lemma 3.2.3. *Consider the functions* (3.30), $\ell = 1, \ldots, \mathcal{L}$, *let Assumption 3.2.1 hold, and consider the N-tuples* $\tau_l \in \mathcal{T}$ *corresponding to the functions* $f_l(\xi)$, *for* $l = 1, 2, 3, \mathcal{L}-2, \mathcal{L}-1, \mathcal{L}$, *which are such that*

$$
\begin{aligned}
&T_1(i) = -1, && i = 1, \ldots, N,\\
&T_2(1) = 1, T_2(i) = -1, && i = 2, \ldots, N,\\
&T_3(2) = 1, T_3(i) = -1, && i = 1, \ldots, N, i \neq 2,\\
&T_{\mathcal{L}-2}(2) = -1, T_{\mathcal{L}-2}(i) = 1, && i = 1, \ldots, N, i \neq 2,\\
&T_{\mathcal{L}-1}(1) = -1, T_{\mathcal{L}-1}(i) = 1, && i = 2, \ldots, N,\\
&T_{\mathcal{L}}(i) = 1, && i = 1, \ldots, N.
\end{aligned}
$$

Then it holds that,

i. $f_1(\xi) \leq f_2(\xi) \leq f_3(\xi) \leq f_\ell(\xi) \leq f_{\mathcal{L}-2}(\xi) \leq f_{\mathcal{L}-1}(\xi) \leq f_{\mathcal{L}}(\xi)$, *for any* $\ell = 4, \ldots, \mathcal{L}-3$.

ii. $\lim_{\xi \to -\infty} \left(f_\ell(\xi) - \left(N - \sum_{j=1}^{N} \tau_{\ell,j} \right) \xi \right) = 0,$ *and*

$\lim_{\xi \to \infty} \left(f_\ell(\xi) - \left(N + \sum_{j=1}^{N} \tau_{\ell,j} \right) \xi \right) = 0,$ *for* $\ell = 1, \ldots, \mathcal{L}$.

iii. the function $f_\ell(\xi)$ *is defined over the real numbers for*

(a) $\xi \neq 0$, *for* $\ell = 1, \ldots, \mathcal{L}$, *if* $\sigma_1 \leq 0$.

(b) $\xi \leq -\sqrt{\sigma_1}$ *and* $\xi \geq \sqrt{\sigma_1}$, *for* $\ell = 1, \ldots, \mathcal{L}$, *if* $\sigma_1 > 0$ *and* $\sigma_1 \neq \sigma_2$.

(c) $\xi \leq -\sqrt{\sigma_1}$ *and* $\xi \geq \sqrt{\sigma_1}$, *for* $\ell = 1$ *and* $\ell = \mathcal{L}$, *and for* $\begin{cases} \xi \neq 0 & \text{if } \sigma_3 \leq 0, \\ \xi \leq -\sqrt{\sigma_3} \text{ and } \xi \geq \sqrt{\sigma_3} & \text{if } \sigma_3 > 0, \end{cases}$
for $\ell = 2, 3, \mathcal{L}-2, \mathcal{L}-1$, *if* $\sigma_1 > 0$ *and* $\sigma_1 = \sigma_2$.

iv. if $\sigma_1 > 0$ *and* $\sigma_1 = \sigma_2$, *and there exist* $\sigma_j > 0$ *such that* $\sigma_j \neq \sigma_i$, *for* $i = 1, \ldots, N$, $j = 1, \ldots, N$, $j \neq i$, *then let* $\bar{\sigma} = \max_j(\sigma_j)$. *No function* $f_\ell(\xi)$, $\ell = 1, \ldots, \mathcal{L}$, *is defined over the real numbers for* $-\sqrt{\bar{\sigma}} < \xi < \sqrt{\bar{\sigma}}$.

v. if $f_\ell(\xi)$ *is defined for* $\xi \neq 0$*, then*

$$\lim_{\xi \to 0^-} f_\ell(\xi) = -1 + \sum_{i=1}^{N} \tau_{\ell,i}\sqrt{-\sigma_i} = \bar{a}_\ell^-, \text{ and}$$

$$\lim_{\xi \to 0^+} f_\ell(\xi) = 1 + \sum_{i=1}^{N} \tau_{\ell,i}\sqrt{-\sigma_i} = \bar{a}_\ell^+.$$

vi. if for all $\xi \neq 0$

$$(N-1) + \frac{|\xi|}{\sqrt{\xi^2+1}} \neq \sum_{i=1}^{N} \frac{|\xi|}{\sqrt{\xi^2 - \sigma_i}},$$

then $f_{\mathcal{L}}(\xi)$ *is strictly monotone for* $\xi < 0$ *and* $f_1(\xi)$ *is strictly monotone for* $\xi > 0$.

Proof. The claims are shown using elementary properties of the function $f_\ell(\xi)$, $\ell = 1, \ldots, \mathcal{L}$, defined in (3.30). Recall that by Assumption 3.2.1 $\sigma_1 \geq \sigma_2 \geq \ldots \geq \sigma_N$.
Item *i.* follows from the definition of T_ℓ, for $l = 1, 2, 3, \mathcal{L}-2, \mathcal{L}-1, \mathcal{L}$, and Assumption 3.2.1.
Item *ii.* follows from (3.30) by noting that

$$\lim_{\xi \to \infty} \left(\sqrt{\xi^2 + c} - \xi \right) = 0, \forall c \in \mathbb{R}, |c| < \infty.$$

Item *iii.* follows by noting that (3.30) is defined over the real numbers if $\xi^2 - \sigma_i \geq 0$, for $i = 1, \ldots, N$, and utilising Assumption 3.2.1 and the definition of T_ℓ, $l = 1, 2, 3, \mathcal{L}-2, \mathcal{L}-1, \mathcal{L}$.
Similarly, item *iv.* is demonstrated by noting that if $\sigma_1 = \sigma_2$ the corresponding terms in the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, may cancel out, which effects the interval over which some functions are defined. If any σ_j, $j = 3, \ldots, N$, are positive and not repeated, then the corresponding terms do not cancel in any of the auxiliary functions. Utilising Assumption 3.2.1, the functions in which all terms corresponding to repeated σ_i, $i = 1, \ldots, N$, cancel out are defined over the real numbers if $\xi^2 - \bar{\sigma} \geq 0$.
Item *v.* follows from (3.30) by noting that (3.25) is such that $\lim_{\xi \to 0^-} \hat{f}(\xi) = -1$, and $\lim_{\xi \to 0^+} \hat{f}(\xi) = 1$.
Finally, item *vi.* is shown by noting that

$$\frac{d}{d\xi} f_\ell(\xi) = (N-1) \pm \frac{\xi}{\sqrt{\xi^2+1}} + \sum_{i=1}^{N} \tau_{\ell,i} \frac{\xi}{\sqrt{\xi^2 - \sigma_i}},$$

and using the definition of T_ℓ, $\ell = 1, \mathcal{L}$. □

3.2.3 Conditions for number and properties of solutions

By Lemma 3.2.2, feedback Nash equilibria of the game (3.14), (3.15), $i = 1, \ldots, N$, can be represented graphically by the intersections of the level a with a set of auxiliary functions (3.30), $\ell = 1, \ldots, \mathcal{L}$. The properties of the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, allow to characterise the possible number and location of intersection points, and hence the number and properties of different feedback Nash equilibria the game admits. Utilising the properties of the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, highlighted in Lemma 3.2.3, sufficient conditions for the game to admit a certain number of feedback Nash equilibrium solutions and properties of the resulting closed-loop dynamics are provided in the following result.

Theorem 3.2.1. *Consider the game* (3.14), (3.15), $i = 1, \ldots, N$, *and let Assumptions 3.2.1 and 3.2.2 hold. Then,*

i. if the open-loop system is unstable with a fast rate of divergence, i.e. (3.14) *is such that* $|a| \gg 1$, *then there exist* $2^N - 1$ *feedback Nash equilibrium solutions.*

ii. if $\sigma_1 > 0$, *then there always exists at least one feedback Nash equilibrium solution for any value of* a.

iii. if $\sigma_1 > 0$ *and there exist* $\sigma_j > 0$ *such that* $\sigma_j \neq \sigma_i$, *for* $i = 1, \ldots, N$, $j = 1, \ldots, N$, $j \neq i$, *and* $\bar{\sigma} = \max_j(\sigma_j)$, *then the system* (3.14) *in closed loop with any Nash equilibrium solution is such that* $|a_{cl}^\star| \leq \left|\sqrt{\bar{\sigma}} - \sqrt{\bar{\sigma} + 1}\right|$.

iv. if $\sigma_N \geq 0$ *and the system* (3.14) *is open-loop stable, i.e.* $|a| < 1$, *then there exists a unique feedback Nash equilibrium solution.*

v. if $\sigma_N < 0$, *but* $\sigma_1 > 0$, $\sigma_1 > \sigma_2$ *and the system* (3.14) *is open-loop stable with a fast rate of convergence, i.e.* $|a| \ll 1$, *then there exists a unique feedback Nash equilibrium solution if* $\sum_{i=2}^{N} \sqrt{\sigma_1 - \sigma_i} < (N-1)\sqrt{\sigma_1} + \sqrt{\sigma_1 + 1}$ *and* $(N-1) + \frac{|\xi|}{\sqrt{\xi^2 + 1}} < \sum_{i=1}^{N} \frac{|\xi|}{\sqrt{\xi^2 - \sigma_i}}$.

vi. if $\sigma_1 \leq 0$, *and the system* (3.14) *is open-loop stable with a fast rate of convergence, i.e.* $|a| \ll 1$, *then there exists a unique feedback Nash equilibrium solution if* $\left(\sum_{i=2}^{N} \sqrt{-\sigma_i}\right) - \sqrt{-\sigma_1} < 1$ *and* $(N-1) + \frac{|\xi|}{\sqrt{\xi^2 + 1}} \neq \sum_{i=1}^{N} \frac{|\xi|}{\sqrt{\xi^2 - \sigma_i}}$.

vii. if $\sigma_1 = \ldots = \sigma_N = 0$ *and* $|a| = 1$ *there exists no feedback Nash equilibrium solution.*

Proof. By Lemma 3.2.2 the intersection points of the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, with the horizontal line at level a characterise distinct feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, \ldots, N$. The claims are then shown by utilising the properties of the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$ in Lemma 3.2.3 to characterise the possible intersection points.
Item *i.* is a result of item *ii.* of Lemma 3.2.3. More precisely, for $\xi < 0$ $f_{\mathcal{L}}(\xi)$ tends to 0 whereas $f_\ell(\xi)$, $\ell = 1, \ldots, \mathcal{L} - 1$, asymptotically approach straight lines with positive slope and hence tend to $-\infty$ as $\xi \to -\infty$. Similarly, for $\xi > 0$ $f_1(\xi)$ tends to 0 and $f_\ell(\xi)$, $\ell = 2, \ldots, \mathcal{L}$, asymptotically approach straight lines with positive slope and hence tend to ∞ as $\xi \to \infty$. Thus, for large values of a, the level a intersects $\mathcal{L} - 1 = 2^N - 1$ of the $\mathcal{L}$ auxiliary functions once.
To demonstrate item *ii.*, recall item *iii.(b),(c)* of Lemma 3.2.3 and note that by definition (3.30) $f_\ell(\xi)$, $\ell = 1, \ldots, \mathcal{L}$, are continuous for $\xi \neq 0$, that $f_{\mathcal{L}-1}(-\sqrt{\sigma_1}) = f_{\mathcal{L}}(-\sqrt{\sigma_1})$, and $f_1(\sqrt{\sigma_1}) = f_2(\sqrt{\sigma_1})$. From item *ii.* of Lemma 3.2.3, for $\ell = 1, 2, \mathcal{L}-1, \mathcal{L}$, it follows that for any value of $a \neq 0$, the horizontal line at level a intersects at least once with either of the functions $f_\ell(\xi)$, $\ell = 1, 2, \mathcal{L} - 1, \mathcal{L}$. If $a = 0$ there

exists at least one feedback Nash equilibrium solution as discussed in Remark 3.2.2.

Item *iii.* follows from item *iv.* of Lemma 3.2.3, namely, if the stated conditions hold any intersection points of the functions (3.30), $\ell = 1, \dots, \mathcal{L}$, with the horizontal line at level a are such that $\xi^\star \leq -\sqrt{\bar{\sigma}}$ or $\xi^\star \geq \sqrt{\bar{\sigma}}$. By recalling from Lemma 3.2.2 that $a_{cl}^\star = \hat{f}(\xi^\star)$, the bound on the closed-loop dynamics follows from the definition of $\hat{f}(\xi)$ in (3.25).

To prove item *iv.*, recall items *i.*, *ii.*, *iii.* and *vi.* of Lemma 3.2.3. If $\sigma_N \geq 0$, then by Assumption 3.2.1 all σ_i, $i = 1, \dots, N$, are non-negative. Hence, the line at level a intersects only one of the functions (3.30), $\ell = 1, \mathcal{L}$, once for $\max(f_{\mathcal{L}-1}(\xi)) < a < \min(f_2(\xi))$. For $\xi > 0$,

$$\begin{aligned} f_2(\xi) &= (N-1)\xi + \sqrt{\xi^2+1} + \sqrt{\xi^2-\sigma_1} - \sum_{i=2}^{N} \sqrt{\xi^2-\sigma_i} \\ &= \sqrt{\xi^2+1} + \sqrt{\xi^2-\sigma_1} + \sum_{i=2}^{N} \frac{\sigma_i}{\xi + \sqrt{\xi^2-\sigma_i}} \\ &\geq \sqrt{\xi^2+1} + \sqrt{\xi^2-\sigma_1} + \frac{1}{2\xi}\sum_{i=2}^{N} \sigma_i := \underline{f}_2(\xi). \end{aligned}$$

If $\xi \geq \sqrt{\sigma_1}$, then $f_2(\xi) \geq \underline{f}_2(\xi) \geq \sqrt{\xi^2+1} + \sum_{i=2}^{N} \frac{\sigma_i}{\xi+\sqrt{\xi^2+\sigma_i}} \geq \sqrt{\xi^2+1} \geq 1$. Recall that if $\sigma_1 > \sigma_2$ then $f_2(\xi)$ is defined for $\xi \geq \sqrt{\sigma_1}$, hence $\min(f_2(\xi)) \geq 1$. If $\sigma_1 = \sigma_2$, then

$$\begin{aligned} f_2(\xi) &= (N-1)\xi + \sqrt{\xi^2+1} - \sum_{N=3}^{N} \sqrt{\xi^2-\sigma_i} \\ &= \xi + \sqrt{\xi^2+1} + \sum_{i=3}^{N} \frac{\sigma_i}{\xi + \sqrt{\xi^2-\sigma_i}} \\ &\geq \sqrt{\xi^2+1} \geq 1, \end{aligned}$$

and hence $\min(f_2(\xi)) \geq 1$. Via analogous arguments for $f_{\mathcal{L}-1}(\xi)$ for $\xi < 0$, it follows that for $-1 < a < 1$ the line at level a intersects only once with either $f_{\mathcal{L}}(\xi)$ for $\xi < 0$ or $f_1(\xi)$ for $\xi > 0$.

Similarly, if the conditions of item *v.* hold, then $f_{\mathcal{L}}(-\sqrt{\sigma_1}) < f_1(\sqrt{\sigma_1})$ and $f_{\mathcal{L}}(\xi)$ for $\xi < 0$ and $f_1(\xi)$ for $\xi > 0$ are strictly decreasing. This follows from the definition of the functions (3.30), and T_ℓ, $\ell = 1, \mathcal{L}$, as well as item *vi.* of Lemma 3.2.3. Hence, by items *i.*, *ii.* and *iii.(b)* of Lemma 3.2.3 the line at level a intersects only once with either $f_{\mathcal{L}}(\xi)$ for $\xi < 0$ or $f_1(\xi)$ for $\xi > 0$ for very small values of a. Likewise, if the conditions of item *vi.* hold then by items *iv.* and *vi.* of Lemma 3.2.3 $\bar{a}_{\mathcal{L}-1}^- < \bar{a}_2^+$ and $f_{\mathcal{L}}(\xi)$ for $\xi < 0$ and $f_1(\xi)$ for $\xi > 0$ are both either strictly decreasing or strictly increasing. Hence, by items *i.*, *ii.* and *iii.(a)* of Lemma 3.2.3 the line at level a intersects only once with either $f_{\mathcal{L}}(\xi)$ for $\xi < 0$ or $f_1(\xi)$ for $\xi > 0$ for very small values of a.

Finally, item *vii.* follows from items *i.*, *ii.* and *v.* of Lemma 3.2.3, by recalling from the proof of item *iv.* that if $\sigma_i \geq 0$, for $i = 1, \dots, N$, then $f_{\mathcal{L}-1}(\xi) \leq -1$ for $\xi < 0$ and $f_2(\xi) \geq 1$ for $\xi > 0$, and by noting that if $\sigma_i = 0$, for $i = 1, \dots, N$, then $\bar{a}_\ell^- = -1$ and $\bar{a}_\ell^+ = 1$ for $\ell = 1, \dots, \mathcal{L}$. Hence, since the functions (3.30), $\ell = 1, \ldots, \mathcal{L}$, are not defined at $\xi = 0$, the horizontal lines $a = 1$ and $a = -1$ do not intersect with any of the auxiliary functions. □

Remark 3.2.4. It is interesting to compare the results of Theorem 3.2.1 with the continuous-time counterpart. While analogous results to items *i.* - *iv.* and *vii.* can be derived for scalar LQ differential games, see [58], note that for this class of continuous-time dynamic games there always exists a unique feedback Nash equilibrium for games involving open-loop stable systems with fast rate of convergence, irrespective of the the signs of σ_i, $i = 1, \ldots, N$, [58, Theorem 3.1]. In the discrete-time setting, however, this only holds true for special cases if $\sigma_N < 0$, such as under the conditions of items *v.* and *vi.* of Theorem 3.2.1.

3.2.4 The two-player case

Consider the game (3.14), (3.15), $i = 1, \ldots, N$, and let $N = 2$. This special case, the class of scalar two-player games, is of particular interest, because additional insights about feedback Nash equilibrium solutions can be gained. Not only is it possible to derive conditions in terms of the system and cost parameters characterising the exact number of feedback Nash equilibria the game admits, but it is also possible to graphically represent the coupled equations characterising feedback Nash equilibria and their solutions in an alternative way via plane curves, and hence to illustrate further properties of different solutions. For ease of exposition, the focus of this subsection lies on the commonly considered case $q_i \geq 0$, for $i = 1, 2$. This is in line with the game definitions considered in the other sections of this chapter and in Chapter 5. While the main results of this subsection, i.e. the minimum and maximum number of feedback Nash equilibria a scalar two-player game may admit and the analysis of the coupled equations characterising solutions via plane curves, also apply to the more general case, the characterisation of conditions becomes significantly more cumbersome if any q_i, $i = 1, 2$, are negative and is hence omitted for clarity.

Characterisation of number of feedback Nash equilibria

In the following, similar arguments as in the context of Theorem 3.2.1 are specialised to the two-player case to provide conditions under which the game (3.14), (3.15), $i = 1, 2$, admits no, a unique, two or three feedback Nash equilibrium solutions. To this end, recall the auxiliary functions (3.30), $\ell = 1, \ldots, 4$, for $N = 2$, and the properties highlighted in Lemma 3.2.3. Moreover, take into account the additional properties of functions $f_2(\xi)$ and $f_3(\xi)$ given in the following result.

Lemma 3.2.4. *Consider the functions* (3.30)*, $\ell = 1, \ldots, 4$, for $N = 2$, and let Assumption 3.2.1 hold. If $\sigma_1 > \sigma_2 \geq 0$, then $f_2(\xi)$ has a unique stationary point corresponding to a local maximum $f_2^\star$ at $\xi_{f_2}^\star < -\sqrt{\sigma_1}$ and $f_3(\xi)$ has a unique stationary point corresponding to a local minimum $f_3^\star$ at $\xi_{f_3}^\star > \sqrt{\sigma_1}$.*

Proof. If $\sigma_1 > \sigma_2 \geq 0$, then by item *iii.(b)* of Lemma 3.2.3 the functions (3.30), $\ell = 1, \dots, 4$, are defined for $\xi \leq -\sqrt{\sigma_1}$ and $\xi \geq \sqrt{\sigma_1}$. Moreover,

$$\begin{aligned}\lim_{\xi \to -\infty} \frac{df_3}{d\xi} &= 0, \quad \lim_{\xi \to -\sqrt{\sigma_1}} \frac{df_3}{d\xi} = \infty,\\ \lim_{\xi \to \infty} \frac{df_3}{d\xi} &= 2, \quad \lim_{\xi \to \sqrt{\sigma_1}} \frac{df_3}{d\xi} = -\infty,\end{aligned}$$

and $\frac{d^2 f_3}{d\xi^2} > 0$ for all ξ, indicating a unique minimum for $\xi > \sqrt{\sigma_1}$. The claim is shown via analogous arguments for $f_2(\xi)$. □

Noting that

$$f_1(-\sqrt{\sigma_1}) = f_2(-\sqrt{\sigma_1}) = -\sqrt{\sigma_1} - \sqrt{\sigma_1 + 1} - \sqrt{\sigma_1 - \sigma_2} := a_1^-, \tag{3.31a}$$

$$f_3(-\sqrt{\sigma_1}) = f_4(-\sqrt{\sigma_1}) = -\sqrt{\sigma_1} - \sqrt{\sigma_1 + 1} + \sqrt{\sigma_1 - \sigma_2} := a_4^-, \tag{3.31b}$$

$$f_1(\sqrt{\sigma_1}) = f_2(\sqrt{\sigma_1}) = \sqrt{\sigma_1} + \sqrt{\sigma_1 + 1} - \sqrt{\sigma_1 - \sigma_2} := a_1^+, \tag{3.31c}$$

$$f_3(\sqrt{\sigma_1}) = f_4(\sqrt{\sigma_1}) = \sqrt{\sigma_1} + \sqrt{\sigma_1 + 1} + \sqrt{\sigma_1 - \sigma_2} := a_4^+, \tag{3.31d}$$

conditions under which the game (3.14), (3.15), $i = 1, 2$, admits no, a unique, two or three feedback Nash equilibrium solutions are provided in the following result.

Theorem 3.2.2. *Consider the game* (3.14), (3.15), $i = 1, 2$. *Let Assumption 3.2.1 hold and consider the functions* (3.30), $\ell = 1, \dots, 4$. *Then,*

i. if $\sigma_1 > \sigma_2 \geq 0$

- *(a) the game has a unique feedback Nash equilibrium solution if* $f_2^\star < a < f_3^\star$,
- *(b) the game has two feedback Nash equilibrium solutions if* $a = f_2^\star$ *or* $a = f_3^\star$,
- *(c) the game has three feedback Nash equilibrium solutions if* $a < f_2^\star$ *or* $a > f_3^\star$,

ii. if $\sigma_1 = \sigma_2 > 0$

- *(a) the game has a unique feedback Nash equilibrium solution if* $a = a_1^-$ *or* $-1 \leq a \leq 1$ *or* $a = a_1^+$,
- *(b) the game has three feedback Nash equilibrium solutions if* $a < a_1^-$ *or* $a_1^- < a < -1$ *or* $1 < a < a_1^+$ *or* $a > a_1^+$,

iii. if $\sigma_1 = \sigma_2 = 0$

- *(a) the game has no feedback Nash equilibrium solution if* $|a| = 1$,
- *(b) the game has a unique feedback Nash equilibrium solution if* $-1 < a < 1$,
- *(c) the game has three feedback Nash equilibrium solutions if* $a < -1$ *or* $a > 1$.

Proof. The claim follows from Lemma 3.2.2 and the properties of the functions (3.30), $\ell = 1, \dots, 4$. To show item *i.*, recall items *i.* *ii.* and *iii.(b)* of Lemma 3.2.3, as well as Lemma 3.2.4. To show items *ii.* and *iii.*, note that if $\sigma_1 = \sigma_2$, then $f_2(\xi) = f_3(\xi)$ and $a_1^- = a_4^-$, $a_1^+ = a_4^+$, and recall items *i.* *ii.* and *iii.(a),(c)* of Lemma 3.2.3. The case $a = 0$ is discussed in Remark 3.2.2. □

Remark 3.2.5. Recall from Remark 3.2.3 that if the line at level a intersects multiple auxiliary functions at a point in which they coincide, this generally indicates distinct feedback Nash equilibrium solutions resulting in the same closed-loop dynamics. In the considered two-player context this is particularly relevant if $\sigma_1 = \sigma_2$, since in this case $f_2(\xi) = f_3(\xi)$. However, if $\sigma_1 > \sigma_2 \geq 0$ as in item *i.* of Theorem 3.2.2, the lines at $a = a_1^-$ and $a = a_1^+$ intersect both $f_1(\xi)$ and $f_2(\xi)$ in the points $(-\sqrt{\sigma_1}, a_1^-)$ and $(\sqrt{\sigma_1}, a_1^+)$, and the lines at $a = a_4^-$ and $a = a_4^+$ intersect both $f_3(\xi)$ and $f_4(\xi)$ in the points $(-\sqrt{\sigma_1}, a_4^-)$ and $(\sqrt{\sigma_1}, a_4^+)$. If $\sigma_1 = \sigma_2 > 0$ as in item *ii.* of Theorem 3.2.2, the lines at $a = a_1^-$ and $a = a_1^+$ even intersect all four auxiliary functions in the points $(-\sqrt{\sigma_1}, a_1^-)$ and $(\sqrt{\sigma_1}, a_1^+)$ (recall that in this case $a_1^- = a_4^-$, $a_1^+ = a_4^+$). However, all of these described intersection points fall under the special case discussed in Remark 3.2.3, hence each of the points indicates one feedback Nash equilibrium, rather than two (if $\sigma_1 > \sigma_2 \geq 0$) or four (if $\sigma_1 = \sigma_2 > 0$) distinct feedback Nash equilibria.

Plane curve interpretation of coupled equations

To gain additional insights regarding the coupled equations (3.18), $i = 1, 2$, and their solutions, the fact that $N = 2$ is utilised to introduce an alternative graphical representation of the conditions as plane curves. To this end, consider the following reformulation of Corollary 3.2.1.

Lemma 3.2.5. *Consider the game* (3.14), (3.15), $i = 1, 2$. *The pair of strategies* $\{\phi_1^\star(x(k)), \phi_2^\star(x(k))\}$, *with* $\phi_i^\star(x(k))$ *given by* (3.16), *for* $i = 1, 2$, *with*

$$k_1^\star = -\frac{r_2 b_1 p_1^\star a}{r_1 r_2 + r_2 b_1^2 p_1^\star + r_1 b_2^2 p_2^\star}\,, \tag{3.32a}$$

$$k_2^\star = -\frac{r_1 b_2 p_2^\star a}{r_1 r_2 + r_2 b_1^2 p_1^\star + r_1 b_2^2 p_2^\star}\,, \tag{3.32b}$$

constitutes a feedback Nash equilibrium of the game if and only if $p_i^\star \in \mathbb{R}$, *for* $i = 1, 2$, *satisfy*

$$\begin{aligned} 0 &= r_2^2 b_1^4 p_1^{\star 3} + 2 r_1 r_2 b_1^2 b_2^2 p_1^{\star 2} p_2^\star + r_1^2 b_2^4 p_1^\star p_2^{\star 2} \\ &\quad + (2 r_1 r_2^2 b_1^2 - q_1 r_2^2 b_1^4 - r_1 r_2^2 b_1^2 a^2) p_1^{\star 2} - q_1 r_1^2 b_2^4 p_2^{\star 2} \\ &\quad + 2(r_1^2 r_2 b_2^2 - r_1 r_2 q_1 b_1^2 b_2^2) p_1^\star p_2^\star - 2 q_1 r_1^2 r_2 b_2^2 p_2^\star \\ &\quad + r_1 r_2^2 (r_1 - a^2 r_1 - 2 q_1 b_1^2) p_1^\star - q_1 r_1^2 r_2^2\,, \end{aligned} \tag{3.33a}$$

$$
\begin{aligned}
0 = {} & r_1^2 b_2^4 p_2^{\star 3} + 2 r_1 r_2 b_1^2 b_2^2 p_2^{\star 2} p_1^\star + r_2^2 b_1^4 p_2^\star p_1^{\star 2} \\
& + (2 r_2 r_1^2 b_2^2 - q_2 r_1^2 b_2^4 - r_2 r_1^2 b_2^2 a^2) p_2^{\star 2} - q_2 r_2^2 b_1^4 p_1^{\star 2} \\
& + 2 (r_2^2 r_1 b_1^2 - r_1 r_2 q_2 b_1^2 b_2^2) p_2^\star p_1^\star - 2 q_2 r_2^2 r_1 b_1^2 p_1^\star \\
& + r_1^2 r_2 (r_2 - a^2 r_2 - 2 q_2 b_2^2) p_2^\star - q_2 r_1^2 r_2^2 \,,
\end{aligned}
\tag{3.33b}
$$

such that either

$$
p_2^\star > -\frac{b_1^2 r_2}{b_2^2 r_1} p_1^\star - \frac{r_2(1 - |a|)}{b_2^2} , \tag{3.34a}
$$

or

$$
p_2^\star < -\frac{b_1^2 r_2}{b_2^2 r_1} p_1^\star - \frac{r_2(1 + |a|)}{b_2^2} . \tag{3.34b}
$$

Proof. The claim follows from Corollary 3.2.1 by eliminating $k_1^\star$, $k_2^\star$ in (3.18) (by solving (3.18b) for $k_i^\star$, $i = 1, 2$ and substituting this into (3.18a)). Hence, any $p_1^\star$, $p_2^\star$ satisfying (3.33) and

$$
r_1 r_2 + r_2 b_1^2 p_1^\star + r_1 b_2^2 p_2^\star \neq 0 \,, \tag{3.35}
$$

also satisfy (3.18) with $k_i^\star = -r_j b_i p_i^\star a / (r_1 r_2 + r_2 b_1^2 p_1^\star + r_1 b_2^2 p_2^\star)$, for $i = 1, 2$, $j = 1, 2$, $j \neq i$, which gives (3.32). Utilising (3.32), the condition (3.34) corresponds to (3.17). Finally, note that (3.34) implies (3.35). □

Lemma 3.2.5 shows that feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, 2$, are characterised by the stabilising solutions of a pair of *cubic* equations. In general, there may be up to nine solutions (including multiplicities) to a pair of cubic equations [174, Chapter 3.11]. However, taking a closer look at equations (3.33) and their characteristics, it is possible to make more precise statements about the number and properties of solutions. To this end, note that the equations (3.33) represent two cubic curves in the (p_1, p_2) plane, whose characteristics are summarised in the following result.

Proposition 3.2.1. *If $p_i^\star \neq q_i$, for $i = 1, 2$, then the solutions $p_1^\star$, $p_2^\star$ of* (3.33a) *and* (3.33b) *are represented by the intersections of the curves*

$$
\tilde{p}_1 = -\frac{b_2^2 r_1}{b_1^2 r_2} p_2 - \frac{r_1}{b_1^2} \pm \frac{a r_1 \sqrt{p_2 r_2^3 \left(p_2 b_2^2 + r_2\right) (p_2 - q_2)}}{b_1^2 r_2^2 (p_2 - q_2)} , \tag{3.36a}
$$

$$
\tilde{p}_2 = -\frac{b_1^2 r_2}{b_2^2 r_1} p_1 - \frac{r_2}{b_2^2} \pm \frac{a r_2 \sqrt{p_1 r_1^3 \left(p_1 b_1^2 + r_1\right) (p_1 - q_1)}}{b_2^2 r_1^2 (p_1 - q_1)} , \tag{3.36b}
$$

in the region of the (p_1, p_2) plane which is such that

$$p_2 > -\frac{b_1^2 r_2}{b_2^2 r_1} p_1 - \frac{r_2(1-|a|)}{b_2^2}, \tag{3.37a}$$

or

$$p_2 < -\frac{b_1^2 r_2}{b_2^2 r_1} p_1 - \frac{r_2(1+|a|)}{b_2^2}. \tag{3.37b}$$

If $q_i \geq 0$, the plane curves (3.36) are each characterised by three branches: a closed branch on the interval $-\frac{r_j}{b_j^2} \leq p_j \leq 0$ and two open branches on the interval $q_j < p_j$. If $p_j < -\frac{r_j}{b_j^2}$ or $0 < p_j \leq q_j$, then $\tilde{p}_i$ in (3.36) is not defined over the real numbers, for $i = 1, 2$, $j = 1, 2$, $j \neq i$. The open branches of $\tilde{p}_i$, $i = 1, 2$, converge to a linear asymptote at

$$p_j = q_j,$$

and a parabolic asymptote described by

$$\lim_{p_j \to \infty} (\tilde{p}_i - \tilde{p}_i^{as}) = 0, \ \tilde{p}_i^{as} = -\frac{b_j^2 r_i}{b_i^2 r_j} p_j - \frac{r_i}{b_i^2} \pm \frac{a r_i}{b_i^2 r_j^2} \sqrt{r_j^3 b_j^2 \left(p_j + \frac{r_j}{b_j^2} + q_j\right)}, \tag{3.38}$$

for $i = 1, 2$, $j = 1, 2$, $j \neq i$.

Proof. Note that (3.36) is obtained by solving (3.33a) for $p_2^\star$ and (3.33b) for $p_1^\star$. This introduces the condition $p_i^\star \neq q_i$, $i = 1, 2$. Any $p_1^\star$, $p_2^\star$ satisfying both equations (3.36) geometrically correspond to the intersections of the plane curves. The region of the (p_1, p_2) plane of interest (3.37) follows from the stability condition (3.34). The function $\tilde{p}_i$ in (3.36) is defined over the real numbers if the argument of the square root is non-negative, i.e. $\hat{p}_i := p_j r_j^3 (p_j b_j^2 + r_j)(p_j - q_j) \geq 0$. If $q_i \geq 0$, this is the case if

$$\begin{cases} p_j \in (q_j, +\infty) & \text{if } p_j > 0, \\ p_j \in \left[-\frac{r_j}{b_j^2},\ 0\right] & \text{otherwise,} \end{cases}$$

for $i = 1, 2$, $j = 1, 2$, $j \neq i$. Noting that both $\tilde{p}_i(-\frac{r_j}{b_j^2})$ and $\tilde{p}_i(0)$ lie on the line

$$\bar{p}_i = -\frac{b_j^2 r_i}{b_i^2 r_j} p_j - \frac{r_i}{b_i^2}, \tag{3.39}$$

and that

$$\tilde{p}_i^+ = \bar{p}_i + \frac{a r_i \sqrt{\hat{p}_i}}{b_i^2 r_j^2 (p_j - q_j)} \geq \bar{p}_i \text{ and } \tilde{p}_i^- = \bar{p}_i - \frac{a r_i \sqrt{\hat{p}_i}}{b_i^2 r_j^2 (p_j - q_j)} \leq \bar{p}_i, \tag{3.40}$$

are continuous over $\left[-\frac{r_j}{b_j^2},\ 0\right]$, this corresponds to the closed-branch. For $p_j > 0$, it is straightforward to see that $\lim_{p_j \to q_j} \tilde{p}_i = \pm\infty$, hence there is a linear asymptote at $p_j = q_j$. To analyse the behaviour of $\tilde{p}_i$ as $p_j \to \infty$, rewrite (3.36) as

$$\tilde{p}_i = \bar{p}_i \pm \frac{ar_i}{b_i^2 r_j^2} \sqrt{r_j^3 b_j^2 \left(p_j + \frac{r_j}{b_j^2} + q_j\right) + \frac{q_j r_j^3 (r_j + b_j^2 q_j)}{p_j - q_j}},$$

for $i = 1, 2$, $j = 1, 2$, $j \neq i$. Noting that $\lim_{p_j \to \infty} \frac{q_j r_j^3 (r_j + b_j^2 q_j)}{p_j - q_j} = 0$ gives (3.38). □

Remark 3.2.6. By Lemma 3.2.5 and Proposition 3.2.1 feedback Nash equilibrium solutions of the scalar two-player game (3.14), (3.15), $i = 1, 2$, are represented geometrically by the intersections of the curves (3.36) in the stable region of the (p_1, p_2) plane. It is interesting to put these results into perspective with respect to the continuous-time counterpart. In contrast to the cubic equations (3.33), the continuous-time equivalents, the coupled algebraic Riccati equations, are *quadratic*. Their solutions are geometrically represented by the intersections of two hyperbolas, and the stability constraint (the continuous-time equivalent to (3.37)) divides the (p_1, p_2) plane into a stable and an anti-stable halfplane [30, Chapter 8.4].

Utilising the results of Lemma 3.2.5 and Proposition 3.2.1, it is possible to deduct the number of admissible solutions of the pair of coupled equations (3.33), and hence the possible number of feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, 2$, as detailed in the following result.

Corollary 3.2.2. *Consider the curves* (3.36) *and let the conditions in Proposition 3.2.1 hold. Then, the following hold*

i. The curves (3.36a) *and* (3.36b) *intersect at least once and at most three times in the region* $p_1 \geq q_1$, $p_2 \geq q_2$.

ii. All of these intersections are such that (3.34a) *holds.*

Hence, there exist at least one feedback Nash equilibrium and at most three feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, 2$.

Proof. Recall that $q_i \geq 0$ and $r_i > 0$, for $i = 1, 2$. By Proposition 3.2.1, the open branches of (3.36) converge asymptotically to (3.38) and $p_j = q_j$, for $i = 1, 2$, $j = 1, 2$, $j \neq i$. Due to the signs of the cost parameters q_i, r_i, the straight line $\bar{p}_i$ as defined in (3.39) has a negative slope, for $i = 1, 2$. Note that $\bar{p}_1$ and $\bar{p}_2$ describe the same line, more precisely the line

$$r_1 r_2 + r_2 b_1^2 p_1 + r_1 b_2^2 p_2 = 0, \tag{3.41}$$

and that the asymptotes (3.38), for $i = 1, 2$, are tilted parabolas centred around this line. Hence, the branches $\tilde{p}_1^+$ for $p_2 \geq q_2$ and $\tilde{p}_2^+$ for $p_1 \geq q_1$ as defined in (3.40) cross at least once. To approach the asymptotes, each plane curve $\tilde{p}_i$, $i = 1, 2$, in (3.36) needs to have at least two inflection points. For the branches $\tilde{p}_1^+$ for $p_2 \geq q_2$ and $\tilde{p}_2^+$ for $p_1 \geq q_1$ to cross more than three times, the curves need to change direction at least twice more. However, by Bézout's theorem [174, Chapter 3.11], a cubic curve can only have up to three real inflection points. Hence, there can at most be three intersections. This proves item *(i)*. Note that since $p_j \neq q_j$, $j = 1, 2$, then by Remark 3.2.7 (see below), $q_j \neq 0$ and the intersection points $(p_1^\star, p_2^\star)$ in the region $p_1 \geq q_1$, $p_2 \geq q_2$ are such that $p_j^\star > 0$, $j = 1, 2$. By Lemma 3.2.5 and Proposition 3.2.1, $(p_1^\star, p_2^\star)$ and the corresponding $k_i^\star = -r_j b_i p_i^\star a/(r_1 r_2 + r_2 b_1^2 p_1^\star + r_1 b_2^2 p_2^\star)$, for $i = 1, 2$, $j = 1, 2$, $j \neq i$, as in (3.32), are such that (3.18) hold, with $p_i^\star > 0$, $q_i > 0$, for $i = 1, 2$. By Lyapunov stability arguments, this guarantees that (3.17) holds. Hence (3.34a) holds, which proves item *(ii)*. By Corollary 3.2.1, $(p_1^\star, p_2^\star)$, $(k_1^\star, k_2^\star)$ satisfying (3.17), (3.18), $i = 1, 2$, correspond to feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, 2$. This gives the final claim. □

In the following remarks, the condition $p_i = q_i$, $i = 1, 2$, underlying Proposition 3.2.1 and Corollary 3.2.2, as well as insights which can be gained about the properties of different feedback Nash equilibrium solutions from the proposed graphical representation in Proposition 3.2.1, and the comparison to the continuous-time case are discussed.

Remark 3.2.7. The conditions of Proposition 3.2.1 exclude cases in which $p_i^\star = q_i$, for $i = 1, 2$. Note that there are only two possible cases for the solution of the pair of equations (3.33) to be such that $p_i^\star = q_i$. Namely, if $q_i = -\frac{r_i}{b_i^2}$ or if $q_i = 0$. Since in this section the focus lies on games in which $q_i \geq 0$, only the latter is examined. If $p_i^\star = q_i = 0$, then solving (3.33) gives three possible solutions for $p_j^\star$, $i = 1, 2$, $j = 1, 2$, $j \neq i$. Namely,

$$\begin{aligned} p_j^0 &= -\frac{r_j}{b_j^2}, \\ p_j^+ &= \frac{-r_j + a^2 r_j + b_j^2 q_j}{2b_j^2} + \frac{\sqrt{(r_j + a^2 r_j + b_j^2 q_j)^2 - 4r_j^2 a^2}}{2b_j^2}, \\ p_j^- &= \frac{-r_j + a^2 r_j + b_j^2 q_j}{2b_j^2} - \frac{\sqrt{(r_j + a^2 r_j + b_j^2 q_j)^2 - 4r_j^2 a^2}}{2b_j^2}. \end{aligned}$$

Note that the point $p_i^\star = 0$, $p_j^\star = p_j^0$ lies on the line (3.41) and can hence be disregarded. Using

$$\sqrt{(r_j + a^2 r_j + b_j^2 q_j)^2 - 4r_j^2 a^2} \geq r_j - r_j a^2 + q_j b_j^2,$$

and

$$\sqrt{(r_j + a^2 r_j + b_j^2 q_j)^2 - 4r_j^2 a^2} \geq -r_j + r_j a^2 + q_j b_j^2,$$

it can be shown that $p_i^\star = 0$, $p_j^\star = p_j^-$ always lies outside the stable region described by (3.34).

Similarly, apart from the special case in which $q_1 = 0$, $q_2 = 0$ and $|a| = 1$, the point $p_i^\star = 0$, $p_j^\star = p_j^+$ always lies within the stable region (3.34). Hence, excluding this special case, if $p_i^\star = q_i = 0$, then the feedback Nash equilibrium solutions of the game (3.14), (3.15), $i = 1, 2$, correspond to the intersections of the curves (3.36) and the point $p_i^\star = 0$, $p_j^\star = p_j^+$, for $i = 1, 2$, $j = 1, 2$, $j \neq i$. Note that if $q_j = 0$, $\tilde{p}_i$ in (3.36) coincides with (3.38), for $i = 1, 2$, $j = 1, 2$, $j \neq i$. By analogous arguments as used in the proof of Corollary 3.2.2, the branch $\tilde{p}_j^+$ (as defined in (3.40)) for $p_i \geq q_i$, can intersect the parabola at most twice. Hence, the conclusion of Corollary 3.2.2, namely that there exist at least one and at most three feedback Nash equilibria of the game (3.14), (3.15), $i = 1, 2$, also holds true if $p_i^\star = q_i = 0$ (excluding the case $q_1 = 0$, $q_2 = 0$, $|a| = 1$). Note that this is in line with the conclusions of Theorem 3.2.2 based on the alternative graphical representation of feedback Nash equilibrium solutions introduced in Section 3.2.2.

Remark 3.2.8. In addition to giving insights regarding the possible number of feedback Nash equilibrium solutions the game (3.14), (3.15), $i = 1, 2$, may admit, the plane curve representation of the equations whose solutions characterise feedback Nash equilibria introduced in Proposition 3.2.1 provides an illustration of the Nash equilibrium outcomes associated with different equilibria, i.e. the cost incurred by each player ($J_i^\star = p_i^\star x(0)^2$, for $i = 1, 2$) at the different solutions.

Remark 3.2.9. It is interesting to note that the results presented in Theorem 3.2.2 and Corollary 3.2.2 (plus the special case discussed in Remark 3.2.7) show that the discrete-time game (3.14), (3.15), $i = 1, 2$, admits between zero and three feedback Nash equilibrium solutions, i.e. the minimum and maximum number of distinct solutions the discrete-time scalar two-player game may admit are the same as for its continuous-time counterpart, see e.g. [30, Chapter 8.4]. This holds despite the fact that the coupled algebraic equations (3.33) are cubic rather than quadratic in the decision variables and that the plane curves (3.36) and auxiliary functions (3.30), $\ell = 1, \dots, 4$, are more involved than those arising in the continuous-time case.

3.2.5 Examples

To illustrate the results of this section, consider the following numerical examples. The efficacy of the results of Section 3.2.3 is demonstrated via three four-player games, before the insights specific to two-player games in Section 3.2.4 are visualised via three two-player examples.

Four-player examples

Consider the game defined by the dynamics (3.14) with $N = 4$ and

$$a = \tilde{a}, \quad b_1 = 1, \quad b_2 = 1, \quad b_3 = 1, \quad b_4 = 1, \tag{3.42}$$

and the cost functionals (3.15), $i = 1, \ldots, 4$, with

$$\begin{aligned} q_1 &= 1, \quad q_2 = 0.5, \quad q_3 = 0.25, \quad q_4 = 0.25, \\ r_1 &= 1, \quad r_2 = 1, \quad r_3 = 1, \quad r_4 = 1, \end{aligned} \tag{3.43}$$

and note that $\sigma_i = q_i$, for $i = 1, \ldots, 4$. Firstly, consider $\tilde{a} = 0.7$, which is such that the system (3.14) is open-loop stable. As illustrated in Figure 3.2, there is only a single intersection point between the yellow horizontal line at $a = 0.7$ and the function $f_1(\xi)$ plotted in red. In line with item *iv.* of Theorem 3.2.1, this single intersection point corresponds to a unique feedback Nash equilibrium solution of the game. The corresponding equilibrium values are reported in Table 3.1. Note that in line with item *iii.* of Theorem 3.2.1

$$|a_{cl}^{\star}| = 0.2197 \leq |\sqrt{\sigma_1} - \sqrt{\sigma_1 + 1}| = 0.4142.$$

Secondly, consider $\tilde{a} = 9$. As illustrated in Figure 3.3, there are 11 intersection points between the yellow horizontal line at $a = 9$ and the functions $f_\ell(\xi)$, $\ell = 2, \ldots, 16$. In line with item *i.* of Theorem 3.2.1, these correspond to $2^4 - 1 = 15$ feedback Nash equilibrium solutions, with the corresponding parameter values listed in Table 3.1. Note that since $\sigma_3 = \sigma_4$ four pairs of auxiliary functions coincide, hence, as highlighted in Remark 3.2.3, the four intersection points with these

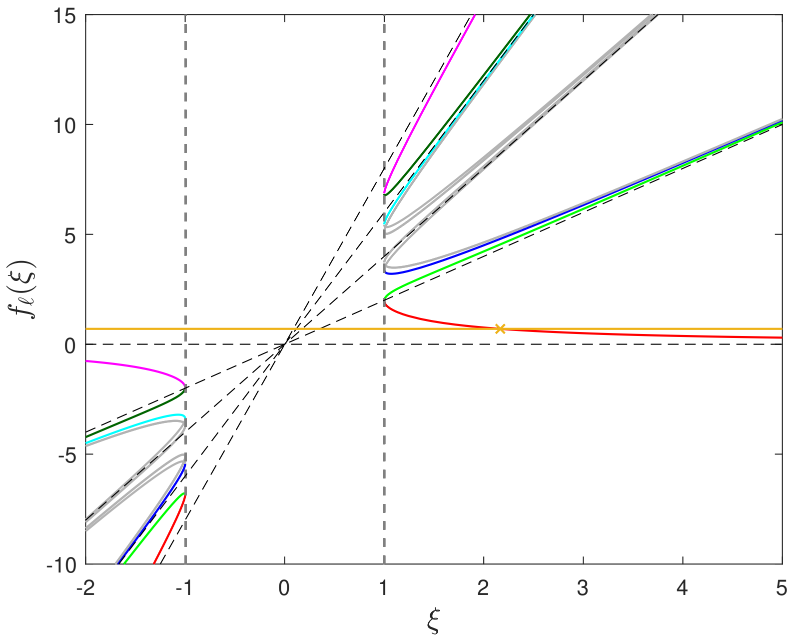


Figure 3.2: Plot of the auxiliary functions $f_1(\xi)$ (red), $f_2(\xi)$ (green), $f_3(\xi)$ (blue), $f_\ell(\xi)$, $\ell = 4, \ldots, 13$ (grey), $f_{14}(\xi)$ (cyan), $f_{15}(\xi)$ (dark green) and $f_{16}(\xi)$ (magenta), and the horizontal line at $a = 0.7$ (yellow). The intersection point of the horizontal line and $f_1(\xi)$ is indicated by the yellow cross. The black dashed lines indicate the linear asymptotes of the auxiliary functions and the grey dotted lines indicate $\xi = \pm\sqrt{\sigma_1} = \pm 1$.

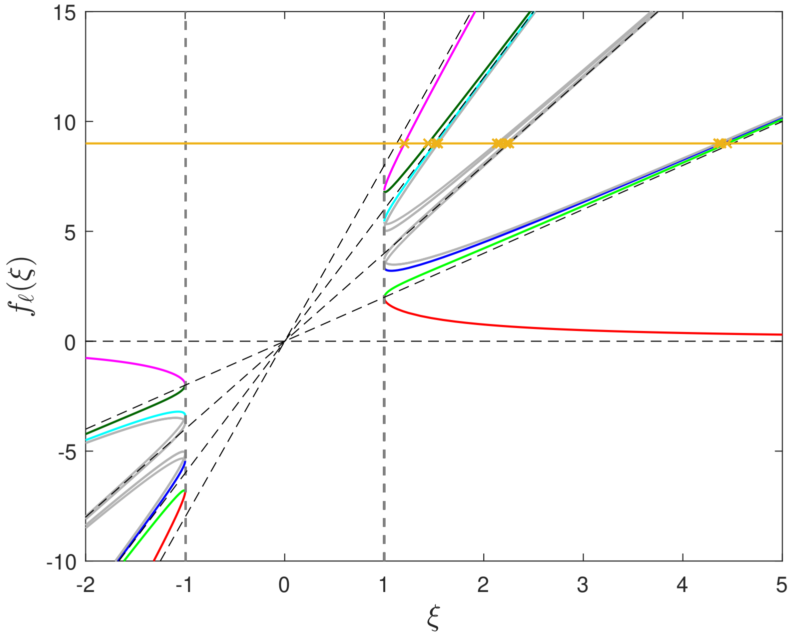


Figure 3.3: Plot of the auxiliary functions $f_1(\xi)$ (red), $f_2(\xi)$ (green), $f_3(\xi)$ (blue), $f_\ell(\xi)$, $\ell = 4, \ldots, 13$ (grey), $f_{14}(\xi)$ (cyan), $f_{15}(\xi)$ (dark green) and $f_{16}(\xi)$ (magenta), and the horizontal line at $a = 9$ (yellow). The intersection points of the horizontal line and $f_\ell(\xi)$, $\ell = 2, \ldots, 16$ are indicated by the yellow crosses. The black dashed lines indicate the linear asymptotes of the auxiliary functions and the grey dotted lines indicate $\xi = \pm\sqrt{\sigma_1} = \pm 1$.

functions correspond to eight feedback Nash equilibria. Note also that as above, the bound $|a_{cl}^{\star}| \leq |\sqrt{\sigma_1} - \sqrt{\sigma_1 + 1}| = 0.4142$ holds for all 15 feedback Nash equilibrium solutions.

Next, consider the game (3.14), (3.15), $i = 1, \ldots, 4$, with a, b_i and r_i, $i = 1, \ldots, 4$, as in (3.42), (3.43) and

$$q_1 = -0.02, \quad q_2 = -0.1, \quad q_3 = -0.15, \quad q_4 = -0.95,$$

and let $\tilde{a} = 0$. Note that for this example, the condition

$$\left(\sum_{i=2}^{N} \sqrt{-\sigma_i} \right) - \sqrt{-\sigma_1} < 1,$$

of item *vi.* of Theorem 3.2.1, which guarantees a unique feedback Nash equilibrium if $|a| \ll 1$, is not satisfied. Indeed, there are six intersections between the yellow horizontal line at level $a = 0$ and the auxiliary functions, as illustrated in Figure 3.4. In line with Remark 3.2.2, the six intersection points indicate seven feedback Nash equilibria of the game. The corresponding equilibrium parameters are listed in Table 3.1. This observation highlights a difference between scalar LQ discrete-time dynamic games and their continuous-time counterpart, scalar LQ differential games, for which a unique feedback Nash equilibrium is guaranteed for any values and signs of q_i, $i = 1, \ldots, 4$, if the open-loop system is stable with a fast rate of convergence [58, Theorem 3.1], as discussed in Remark 3.2.4.

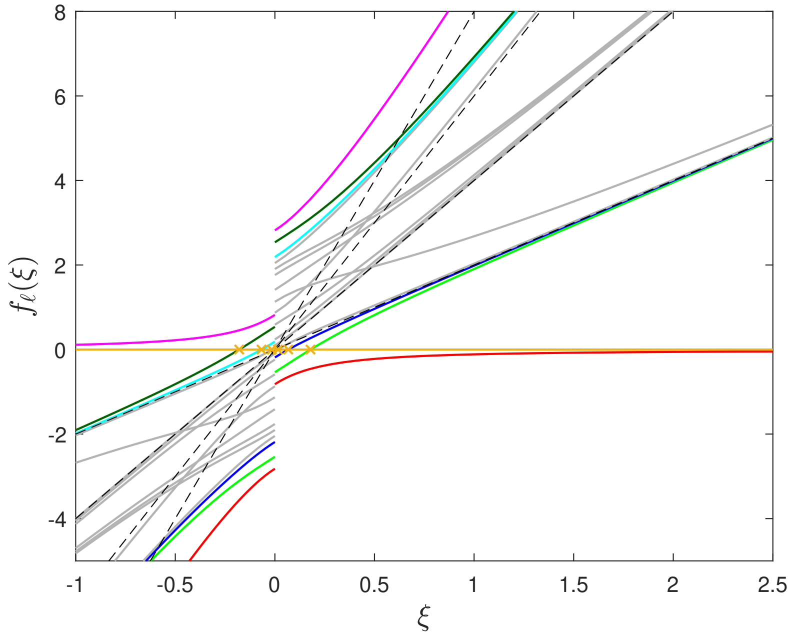


Figure 3.4: Plot of the auxiliary functions $f_1(\xi)$ (red), $f_2(\xi)$ (green), $f_3(\xi)$ (blue), $f_\ell(\xi)$, $\ell = 4, \ldots, 13$ (grey), $f_{14}(\xi)$ (cyan), $f_{15}(\xi)$ (dark green) and $f_{16}(\xi)$ (magenta), and the horizontal line at $a = 0$ (yellow). The intersection points of the horizontal line and $f_\ell(\xi)$, $\ell = 1, \ldots, 16$ are indicated by the yellow crosses. The black dashed lines indicate the linear asymptotes of the auxiliary functions.

Table 3.1: Parameters characterising the feedback Nash equilibrium solutions of the scalar four-player games rounded to four decimal places.

$\tilde{a}$	σ_1	σ_2	σ_3	σ_4	$\xi^\star$	$a^\star_{cl}$	$k^\star_1$	$k^\star_2$	$k^\star_3$	$k^\star_4$
0.7	1	0.5	0.25	0.25	2.1661	0.2197	−0.2446	−0.1187	−0.0585	−0.0585
9	1	0.5	0.25	0.25	1.2026	0.3615	−1.8706	−2.1753	−2.2963	−2.2963
					1.4414	0.3129	−0.4033	−2.6973	−2.7932	−2.7932
					1.5120	0.3008	−2.6460	−0.1755	−2.9388	−2.9388
					1.5407	0.2961	−2.7128	−2.9096	−2.9981	−0.0834
					1.5407	0.2961	−2.7128	−2.9096	−0.0834	−2.9981
					2.1315	0.2229	−0.2491	−0.1207	−4.2036	−4.2036
					2.1635	0.2199	−0.2450	−4.2081	−4.2684	−0.0586
					2.1635	0.2199	−0.2450	−4.2081	−0.0586	−4.2684
					2.2269	0.2142	−4.2167	−0.1152	−4.3970	−0.0569
					2.2269	0.2142	−4.2167	−0.1152	−0.0569	−4.3970
					2.2559	0.2117	−4.2780	−4.3981	−0.0561	−0.0561
					4.3563	0.1133	−0.1163	−0.0578	−8.6838	−0.0288
					4.3563	0.1133	−0.1163	−0.0578	−0.0288	−8.6838
					4.3861	0.1126	−0.1155	−8.7147	−0.0286	−0.0286
					4.4449	0.1111	−8.7759	−0.0566	−0.0282	−0.0282
0	−0.02	−0.1	−0.15	−0.95	−0.1790	−0.8369	0.4071	−0.1844	−0.2477	−0.8120
					−0.0670	−0.9352	−0.0895	0.3903	−0.3260	−0.9100
					−0.0153	−0.9848	−0.1269	−0.3013	0.4029	−0.9595
					0.0153	0.9848	0.1269	0.3013	−0.4029	0.9595
					0.0670	0.9352	0.0895	−0.3903	0.3260	0.9100
					0.1790	0.8369	−0.4071	0.1844	0.2477	0.8120
					n/a	0	0	0	0	0

Two-player examples

Consider the game defined by the dynamics (3.14) with $N = 2$ and

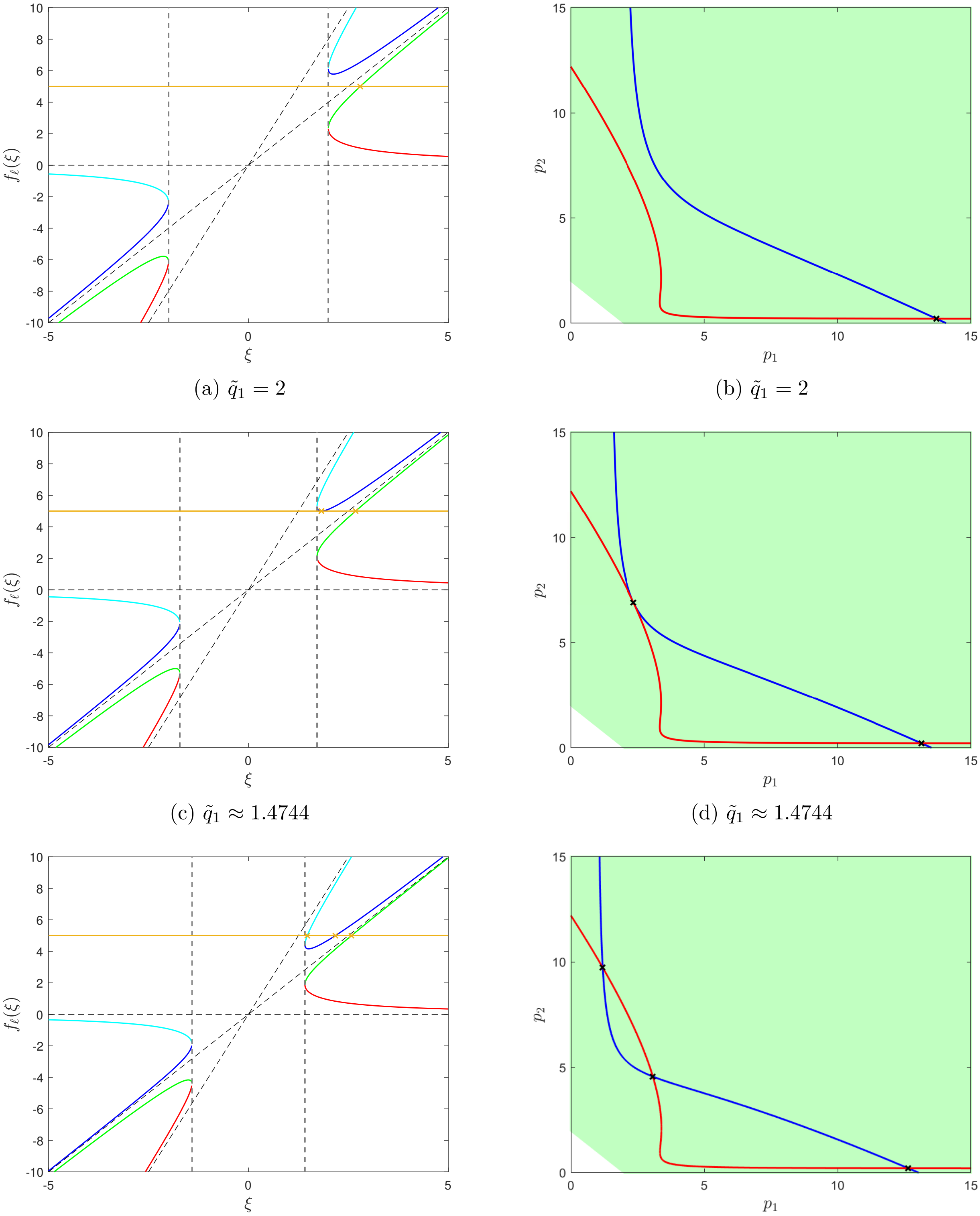
$$a = 5, \quad b_1 = 2, \quad b_2 = 1,$$

and the the cost functionals (3.15), $i = 1, 2$, with

$$\begin{aligned} q_1 &= \tilde{q}_1, \quad q_2 = \frac{1}{5} \\ r_1 &= 2, \quad r_2 = \frac{1}{2}. \end{aligned}$$

Three different cases are considered: $\tilde{q}_1 = 2$, $\tilde{q}_1 = \frac{6640112984933283}{4503599627370496} \approx 1.4744$, and $\tilde{q}_1 = 1$. The corresponding values of σ_i, $i = 1, 2$, are listed in Table 3.2. Firstly, let $\tilde{q}_1 = 2$. The corresponding functions $f_1(\xi)$, $f_2(\xi)$, $f_3(\xi)$ and $f_4(\xi)$ as defined in (3.30) are plotted in solid red, green, blue and cyan, respectively, in Figure 3.5 (a). The linear asymptotes of the functions are depicted with dashed black lines. The level $a = 5$ is highlighted by the horizontal yellow line. There is a single intersection point between this line and $f_2(\xi)$, corresponding to a unique Nash equilibrium solution, which is characterised by the parameters reported in Table 3.2. Next, let $\tilde{q}_1 = \frac{6640112984933283}{4503599627370496}$. The corresponding auxiliary functions (3.30), $\ell = 1, \ldots, 4$, are shown in Figure 3.5 (c). For this specific value of $\tilde{q}_1$, the horizontal line at level $a = 5$ intersects once with $f_2(\xi)$ (green) and intersects $f_3(\xi)$ (blue) at its minimum $f_3^\star$. In line with item *i.(b)* of Theorem 3.2.2, these intersections indicate two feedback Nash equilibrium solutions, with the corresponding values given in Table 3.2. Finally, let $\tilde{q}_1 = 1$. The functions (3.30), $\ell = 1, \ldots, 4$, are shown in Figure 3.5 (e). The line at level $a = 5$ has one intersection point each with the functions $f_2(\xi)$ (green), $f_3(\xi)$ (blue) and $f_4(\xi)$ (cyan). The three intersection points correspond to three feedback Nash equilibrium solutions characterised by the parameters listed in Table 3.2.

Since $N = 2$, it is also possible to represent the coupled equations (3.18), $i = 1, 2$, whose stabilising solutions characterise feedback Nash equilibria via cubic curves in the (p_1, p_2) plane. The plane curves (3.36a) (red) and (3.36b) (blue) for $\tilde{q}_1 = 1$ are shown in Figure 3.6. As stated in Proposition 3.2.1, the open branches of the curves each converge to a linear asymptote (black dotted) and a parabolic asymptote (black dashed). The curves (3.36a) and (3.36b) intersect a total of seven times, however, four intersection points lie outside the stability region (3.37) (highlighted in light green), including two on the line (3.39) (dashed grey line), which are not highlighted because solutions on the line (3.39) are excluded by definition in Lemma 3.2.5. In line with Corollary 3.2.2 and the observations from the auxiliary function representation for this example (see Figure 3.5 (a)), there are three intersections in the region $p_1 \geq q_1$, $p_2 \geq q_2$, corresponding to three feedback Nash equilibrium solutions characterised by the parameters given in Table 3.2. For the three considered values of $\tilde{q}_1$, $\tilde{q}_1 = 2$, $\tilde{q}_1 \approx 1.4744$, and $\tilde{q}_1 = 1$, the intersections of the curves (3.36a) (red) and (3.36b) (blue) in the region $p_1 \geq q_1$, $p_2 \geq q_2$



(a) $\tilde{q}_1 = 2$

(b) $\tilde{q}_1 = 2$

(c) $\tilde{q}_1 \approx 1.4744$

(d) $\tilde{q}_1 \approx 1.4744$

(e) $\tilde{q}_1 = 1$

(f) $\tilde{q}_1 = 1$

Figure 3.5: Graphical interpretation of the coupled equations (3.18), $i = 1, 2$, characterising feedback Nash equilibria and their solutions via auxiliary functions (a), (c), (e) and via cubic plane curves (b), (d), (f). The auxiliary functions $f_1(\xi)$, $f_2(\xi)$, $f_3(\xi)$, and $f_{14}(\xi)$ are plotted in red, green, blue and cyan, respectively, and the horizontal line at $a = 5$ is plotted in yellow. The intersection points are indicated by the yellow crosses. The black dashed lines indicate the linear asymptotes of the functions and the grey dotted lines indicate $\xi = \pm\sqrt{\sigma_1}$. The cubic plane curves (3.36a) and (3.36b) are plotted in red and blue, respectively, their intersections are highlighted by black crosses and the stability region (3.37) is highlighted in light green.

are shown in more detail in Figure 3.5 (b), (d), and (f), respectively. An advantage of the plane curve representation of the conditions characterising feedback Nash equilibria is that it readily illustrates the equilibrium outcomes associated with different solutions. For instance, for $\tilde{q}_1 = 1$ it is immediately evident from Figures 3.5 (f) and 3.6 that out of the three feedback Nash equilibrium solutions, (NE1) leads to the lowest cost for player 1, and the highest cost for player 2, (NE2) results in a medium cost for both players and (NE3) results in the highest cost for player 1 and the lowest cost for player 2. Similarly, for $\tilde{q}_1 \approx 1.4744$ it is evident from Figure 3.5 (d) that (NEa) leads to a lower cost for player 1 and a higher cost for player 2, whereas (NEb) leads to a lower cost for player 2 at the expense of a higher cost for player 1.

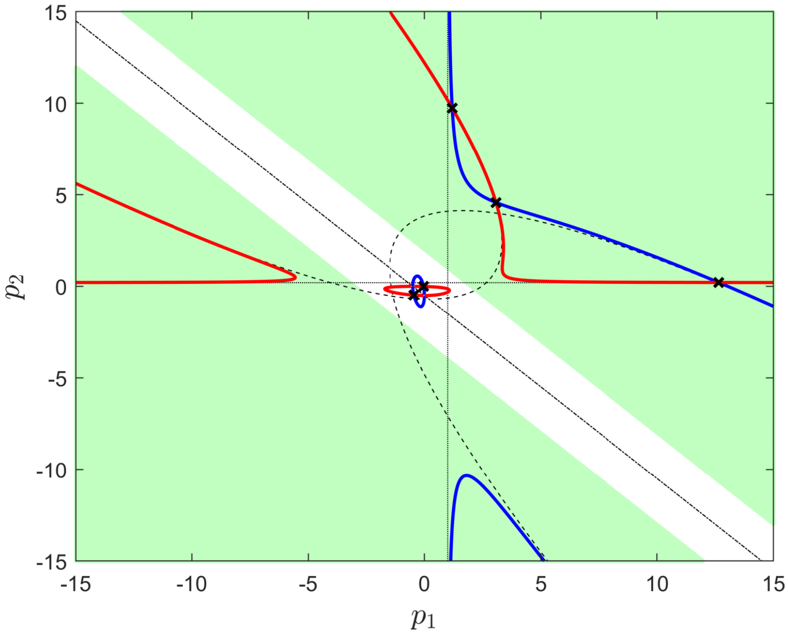


Figure 3.6: The cubic plane curves (3.36a) (red) and (3.36b) (blue), for $\tilde{q}_1 = 1$, their asymptotes (dotted black lines, dashed black curves) and the line (3.39) (dash-dotted black). The stability regions (3.37) are highlighted in light green.

Table 3.2: Parameters characterising the feedback Nash equilibrium solutions of the scalar two-player games. All values rounded to four decimal places.

$\tilde{q}_1$	σ_1	σ_2	$\xi^\star$	$a_{cl}^\star$	$k_1^\star$	$k_2^\star$	$p_1^\star$	$p_2^\star$	
2	4	0.4	2.7978	0.1733	−2.3771	−0.0724	13.7134	0.2089	
1.4744	2.9488	0.4	1.8258	0.2559	−0.6027	−3.5386	2.3552	6.9138	(NEa)
			2.6828	0.1803	−2.3720	−0.0756	13.1552	0.2097	(NEb)
1	2	0.4	1.4737	0.3073	−0.9440	−2.8047	3.0724	4.5641	(NE1)
			2.1766	0.2187	−0.2610	−4.2592	1.1934	9.7363	(NE2)
			2.5783	0.1871	−2.3670	−0.0788	12.6488	0.2105	(NE3)

3.3 Approximate Nash equilibria

Motivated by the challenges associated with determining Nash equilibrium solutions for infinite-horizon nonzero-sum dynamic games, even in the LQ case and particularly in the discrete-time setting, as highlighted in Section 3.1, this section considers a weaker notion of solutions - approximate Nash equilibrium solutions. A well studied notion of approximate Nash equilibrium, in particular in the context of static games, is the ϵ-Nash equilibrium, see e.g. [9, Chapter 4]. In [60] the notion of ϵ-Nash equilibrium is defined in the context of differential games, as recalled in Chapter 2. Related notions of approximate Nash equilibrium solutions for LQ differential games, which can be characterised in terms of matrix inequalities, are introduced in [61, 62]. However, the existing literature in this context focuses on continuous-time dynamic games (i.e. *differential games*).

In this section, a notion of approximate feedback Nash equilibrium solution for discrete-time, infinite-horizon, LQ games is introduced, namely the $\epsilon_{\alpha,\beta}$-Nash equilibrium. The proposed solution concept provides guarantees on the convergence rate of the trajectories of the resulting closed-loop system, which may be of practical importance. The degree of approximation and the computation of the approximate equilibria is discussed. For specific parameter choices, the presented solution concept is the discrete-time counterpart to the concepts presented for continuous-time problems in [61, 62]. However, in addition to focusing on discrete-time systems, the presented formulation includes some additional generalisations with respect to the notions in [61, 62].

The remainder of this section is structured as follows. The considered problem is defined in Section 3.3.1. In Section 3.3.2 the aforementioned notion of approximate feedback Nash equilibrium for the considered class of games is presented. The characterisation of the approximate Nash equilibria via matrix inequalities is proposed and the degree of approximation is discussed in Section 3.3.3. Reformulations of the presented conditions which may aid the computation of solutions are provided in Section 3.3.4. The efficacy of the results is demonstrated via a simulation example, involving a macroeconomic policy design problem, in Section 3.3.5.

3.3.1 Problem formulation

Consider the LQ infinite-horizon discrete-time dynamic game (3.1), (3.2), $i = 1, \ldots, N$, and the problem of determining feedback Nash equilibrium solutions, i.e. feedback strategies of the form $\phi_i(x(k)) = K_i x(k)$ for the players' actions u_i, $i = 1, \ldots, N$, such that (2.8) holds (see Definition 2.1.2). Let Assumption 3.1.1 hold and recall the definition of admissible strategies (Definition 2.1.1), which is refined below in the context of feedback Nash equilibria for the considered class of games.

Definition 3.3.1. A set of feedback strategies $\{\phi_1(x(k)), \ldots, \phi_N(x(k))\}$, where $\phi_i(x(k)) = K_i x(k)$, $i = 1, \ldots, N$, is admissible if the set of control actions $\{u_1, \ldots, u_N\}$, with $u_i = \phi_i(x(k))$, for $i =$

$1,\dots,N$, renders the zero equilibrium of the system (3.1) asymptotically stable.

In the following definition, the notion of α-admissible strategies, which was introduced for continuous-time dynamics in [61], is introduced for the discrete-time system (3.1).

Definition 3.3.2. A set of feedback strategies $\{\phi_1(x(k)),\dots,\phi_N(x(k))\}$, where $\phi_i(x(k)) = K_i x(k)$, $i = 1,\dots,N$, is α-admissible[5] if the set of control actions $\{u_1,\dots,u_N\}$, with $u_i = \phi_i(x(k))$, for $i = 1,\dots,N$, renders the zero equilibrium of the system (3.1) asymptotically stable, and the trajectories of the resulting closed-loop system converge faster than those of the system $x(k+1) = \frac{1}{\alpha} I_n x(k)$, i.e. $\rho(\alpha A_{cl}) < 1$, with $\alpha \geq 1$ and where $A_{cl} = A + \sum_{i=1}^{N} B_i K_i$ is the dynamics matrix of the closed-loop system.

Focusing on admissible strategies in the sense of Definition 3.3.1, Nash equilibria of the game (3.1), (3.2), $i = 1,\dots,N$, are characterised by the stabilising solutions of a set of coupled algebraic equations, (3.5), $i = 1,\dots,N$, and (3.6), see Theorem 3.1.1. Solving (3.5), $i = 1,\dots,N$, and (3.6), and thereby determining feedback Nash equilibrium solutions of the game (3.1), (3.2), $i = 1,\dots,N$, is generally challenging, see e.g. Remark 3.1.2. Hence, the problem considered in this section is to characterise a notion of approximate feedback Nash equilibrium for the game (3.1), (3.2), $i = 1,\dots,N$, which is easier to compute than an exact feedback Nash equilibrium.

3.3.2 A notion of approximate feedback Nash equilibrium

To introduce a notion of approximate Nash equilibrium for LQ infinite-horizon discrete-time nonzero-sum dynamic games observe that the problem of solving the coupled algebraic equations (3.5), $i = 1,\dots,N$, and (3.6), whose stabilising solutions characterise feedback Nash equilibrium solutions for the game (3.1), (3.2), $i = 1,\dots,N$, can be reformulated as a non-convex semidefinite programme (SDP).

Lemma 3.3.1. *Suppose $\mathcal{K}^\star = \{K_1^\star,\dots,K_N^\star\}$, $\mathcal{P}^\star = \{P_1^\star,\dots,P_N^\star\}$ and $\gamma^\star$ constitute a solution of the optimisation problem*

$$\min_{\mathcal{K},\mathcal{P},\gamma} \quad \gamma$$

$$\text{s.t.} \quad P_i \succeq Q_i + \sum_{j=1}^{N} K_j{}^\top R_{ij} K_j + \left(A + \sum_{j=1}^{N} B_j K_j\right)^\top P_i \left(A + \sum_{j=1}^{N} B_j K_j\right), \tag{3.44a}$$

for $i = 1,\dots,N$,

$$\begin{bmatrix} R_{11} + B_1^\top P_1 B_1 & \dots & B_1^\top P_1 B_N \\ \vdots & \ddots & \vdots \\ B_N^\top P_N B_1 & \dots & R_{NN} + B_N^\top P_N B_N \end{bmatrix} \begin{bmatrix} K_1 \\ \vdots \\ K_N \end{bmatrix} = - \begin{bmatrix} B_1^\top P_1 \\ \vdots \\ B_N^\top P_N \end{bmatrix} A, \tag{3.44b}$$

[5] Note that if a set of strategies is α-admissible with $\alpha = 1$, it is admissible.

$$\gamma I_n - \sum_{j=1}^{N} P_j \succeq 0, \tag{3.44c}$$

Then, $\mathcal{K}^\star$*,* $\mathcal{P}^\star$ *constitute a solution of* (3.5)*,* $i = 1, \ldots, N$*, and* (3.6)*.*

Proof. Note that (3.44b) corresponds to (3.6) and that γ subject to (3.44c) and (3.44a) is minimised if (3.44a) holds with equality, for $i = 1, \ldots, N$. Hence, any $\mathcal{K}^\star$, $\mathcal{P}^\star$ solving (3.44) satisfy (3.5), $i = 1, \ldots, N$, (3.6). □

In Lemma 3.3.1 the coupled algebraic matrix equations (3.5), $i = 1, \ldots, N$, and (3.6) are replaced with a minimisation problem subject to matrix inequality and equality constraints, which constitutes a non-convex SDP. While solving (3.44) is still challenging, finding $\{K_1, \ldots, K_N\}$ and $\{P_1, \ldots, P_N\}$ satisfying the matrix inequalities (3.44a), $i = 1, \ldots, N$, may be significantly easier than solving the matrix equations (3.5), $i = 1, \ldots, N$, and is hence of practical interest [9]. Motivated by this, a weaker notion of solution to the game (3.1), (3.2), $i = 1, \ldots, N$, is proposed, which represents a relaxation of the stricter notion of Nash equilibrium solution.

Definition 3.3.3. A set of β-admissible strategies $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$, where $\phi_i^*(x(k)) = K_i^* x(k)$, $i = 1, \ldots, N$, constitutes an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \ldots, N$, if there exists a constant $\epsilon_{x_0,\alpha,\beta} \geq 0$, parameterised in the initial condition $x(0) = x_0$, and $\alpha > 1$, $\beta \geq 1$, such that

$$J_i\left(x(0), \phi_1^*, \ldots, \phi_N^*\right) \leq J_i\left(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right) + \epsilon_{x_0,\alpha,\beta}, \tag{3.45}$$

for all α-admissible $\left\{\phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right\}$, where $\hat{\phi}_i(x(k)) = \hat{K}_i x(k)$, for $i = 1, \ldots, N$.

The considered notion of $\epsilon_{\alpha,\beta}$-Nash equilibrium is related to the notions of ϵ_α-Nash equilibrium introduced in [61, 62] for the continuous-time case. The parameter β introduced herein allows to determine equilibrium solutions that guarantee a "convergence rate" faster than $\frac{1}{\beta}$ of the system (3.1) in closed loop with the strategies $\{K_1^* x(k), \ldots, K_N^* x(k)\}$. Thus, the parameter β can be utilised to impose a certain desired convergence rate on the closed-loop system, which may be of practical interest. In the special cases in which $\beta = 1$ and $\beta = \alpha$ the notion of $\epsilon_{\alpha,\beta}$-Nash equilibrium in Definition 3.3.3 is the *discrete-time equivalent* of the notions in [61] and [62], respectively.

3.3.3 Characterisation via matrix inequalities

In the following, constructive sufficient conditions are provided for determining an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution for an N-player nonzero-sum LQ infinite-horizon discrete-time dynamic game.

Theorem 3.3.1. *Consider the game (3.1), (3.2), $i = 1, \ldots, N$, and constant $\beta \geq 1$. Assume the pair $\left(A, \sum_{i=1}^{N} Q_i\right)$ is detectable. If there exist matrices K_i^*, $P_i^* = P_i^{*\top} \succeq 0$, satisfying the inequality*

$$\Upsilon_i = \frac{1}{\beta^2} P_i^* - Q_i - \sum_{j=1}^{N} K_j^{*\top} R_{ij} K_j^* - \left(A + \sum_{j=1}^{N} B_j K_j^*\right)^\top P_i^* \left(A + \sum_{j=1}^{N} B_j K_j^*\right) \succeq 0, \tag{3.46}$$

for $i = 1, \ldots, N$, and (3.44b) with $K_i = K_i^*$ and $P_i = P_i^*$, then the set of strategies

$$\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\},$$

where

$$\phi_i^*(x(k)) = K_i^* x(k), \tag{3.47}$$

for $i = 1, \ldots, N$, is β-admissible and constitutes an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution of the game, for any $\alpha > 1$. The $\epsilon_{\alpha,\beta}$-Nash equilibrium is such that the equilibrium cost incurred by player i starting from initial condition $x(0)$ is $J_i^* = J_i(x(0), \phi_1^*(x(k)), \ldots, \phi_N^*(x(k))) = x(0)^\top W_i^* x(0)$, where $W_i^* = P_i^* - \Delta W_i$ with ΔW_i satisfying the Lyapunov equation

$$\Delta W_i - \left(A + \sum_{j=1}^{N} B_j K_j^*\right)^\top \Delta W_i \left(A + \sum_{j=1}^{N} B_j K_j^*\right) = \tilde{\Upsilon}_i, \tag{3.48}$$

and $\tilde{\Upsilon}_i = \Upsilon_i + P_i^* - \frac{1}{\beta^2} P_i^*$.

Proof. The proof of the claim consists in showing that the set of strategies $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$ is β-admissible, that it is such that it satisfies (3.45) for all α-admissible $\left\{\phi_1^*(x(k)), \ldots, \phi_{i-1}^*(x(k)), \hat{\phi}_i(x(k)), \phi_{i+1}^*(x(k)), \ldots, \phi_N^*(x(k))\right\}$, where $\hat{\phi}_i(x(k)) = \hat{K}_i x(k)$, for $i = 1, \ldots, N$, and that the resulting cost incurred by player i is $J_i^* = x(0)^\top W_i^* x(0)$, for $i = 1, \ldots, N$. To show the former, consider the sum over i of the inequalities (3.46), for $i = 1, \ldots, N$, namely consider

$$\sum_{i=1}^{N} \Upsilon_i = \frac{1}{\beta^2} \sum_{i=1}^{N} P_i^* - \sum_{i=1}^{N} Q_i - \sum_{i=1}^{N} \sum_{j=1}^{N} K_j^{*\top} R_{ij} K_j^* - \left(A + \sum_{j=1}^{N} B_j K_j^*\right)^\top \sum_{i=1}^{N} P_i^* \left(A + \sum_{j=1}^{N} B_j K_j^*\right) \succeq 0. \tag{3.49}$$

Letting $\tilde{P} = \frac{1}{\beta^2} \sum_{i=1}^{N} P_i^*$ and $A_{cl}^* = A + \sum_{j=1}^{N} B_j K_j^*$, (3.49) implies

$$\tilde{P} - (\beta A_{cl}^*)^\top \tilde{P} (\beta A_{cl}^*) \succeq \sum_{i=1}^{N} Q_i + \sum_{i=1}^{N} \sum_{j=1}^{N} K_j^{*\top} R_{ij} K_j^* \succeq 0. \tag{3.50}$$

Recall that by assumption $\left(A, \sum_{i=1}^{N} Q_i\right)$ is detectable and note that this implies that $\left(\beta A, \sum_{i=1}^{N} Q_i\right)$ is detectable. Assume there exists an eigenvalue $|\lambda| \geq 1$ of βA_{cl}^* and let $v \in \mathbb{R}^n$ be a corresponding eigenvector, i.e. $\beta A^*_{cl} v = \lambda v$. Pre- and postmulitplying (3.50) by v^{H} and v, respectively, gives

$$\left(1 - \lambda^{\mathsf{H}}\lambda\right) v^{\mathsf{H}} \tilde{P} v \succeq v^{\mathsf{H}} \left(\sum_{i=1}^{N} Q_i\right) v + \sum_{j=1}^{N} v^{\mathsf{H}} K_j^{*\top} \left(\sum_{i=1}^{N} R_{ij}\right) K_j^* v \succeq 0.$$

Since $\lambda^{\mathsf{H}}\lambda \geq 1$ and since, by definition, $\tilde{P} \succeq 0$, $\sum_{i=1}^{N} Q_i \succeq 0$ and $\sum_{i=1}^{N} R_{ij} \succ 0$, this implies $v^{\mathsf{H}}\left(\sum_{i=1}^{N} Q_i\right) v = 0$ and $\sum_{j=1}^{N} v^{\mathsf{H}} K_j^{*\top}\left(\sum_{i=1}^{N} R_{ij}\right) K_j^* v = 0$. The former gives $\sum_{i=1}^{N} Q_i v = 0$, and the latter implies $K_j^* v = 0$, for $j = 1, \ldots, N$, hence, $\lambda v = \beta\left(A + \sum_{j=1}^{N} B_j K_j^*\right) v = (\beta A) v$. That is, λ is an undetectable eigenvalue of (βA). This contradicts the original assumption that $\left(\beta A, \sum_{i=1}^{N} Q_i\right)$ is detectable, hence it follows that $\rho\left(\beta A^*_{cl}\right) < 1$, which by Definition 3.3.2 implies β-admissibility. To demonstrate the second claim, note that P_i^*, K_i^* satisfying (3.46), for $i = 1, \ldots, N$, and (3.44b) with $K_i = K_i^*$ and $P_i = P_i^*$ correspond to a Nash equilibrium solution of the dynamic game defined by (3.1) and the modified cost functional

$$\tilde{J}_i(x(0), u_1(\cdot), \ldots, u_N(\cdot)) = \sum_{k=0}^{\infty} \left(x(k)^\top \left(Q_i + \tilde{\Upsilon}_i\right) x(k) + \sum_{j=1}^{N} u_j(k)^\top R_{ij} u_j(k) \right), \tag{3.51}$$

for $i = 1, \ldots, N$. From (3.51) and Definition 2.1.2 it follows that

$$J_i(x(0), \phi_1^*, \ldots, \phi_N^*) \leq \tilde{J}_i(x(0), \phi_1^*, \ldots, \phi_N^*) \leq \tilde{J}_i(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*),$$

for $i = 1, \ldots, N$, and

$$\tilde{J}_i(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*) = J_i(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*) + \sum_{k=0}^{\infty} \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k),$$

for any admissible set of strategies $\left\{\phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right\}$, $i = 1, \ldots, N$, and hence for all α-admissible $\left\{\phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right\}$, $i = 1, \ldots, N$, where $\hat{x}(k)$ satisfies

$$\hat{x}(k+1) = \left(A + \sum_{\substack{j=1,\\ j \neq i}}^{N} B_j K_j^* + B_i \hat{K}_i \right) \hat{x}(k) = \hat{A}_{cl} \hat{x}(k). \tag{3.52}$$

Hence,

$$J_i(x(0), \phi_1^*, \ldots, \phi_N^*) \leq J_i(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*) + \sum_{k=0}^{\infty} \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k). \tag{3.53}$$

The final term in (3.53) can be rewritten as $\sum_{k=0}^{\infty} \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k) = \hat{x}(0)^\top P_{i,\epsilon} \hat{x}(0)$, where

$$P_{i,\epsilon} - \hat{A}_{cl}^\top P_{i,\epsilon} \hat{A}_{cl} = \tilde{\Upsilon}_i,$$

using the definition of the controllability Gramian. Utilising α-admissibility and recalling that $\alpha > 1$, the term $\hat{x}(0)^\top P_{i,\epsilon}\hat{x}(0)$, $i = 1, \ldots, N$, can be upper-bounded by a constant for any $\beta \geq 1$ and any initial condition $\hat{x}(0) = x(0)$. Thus, (3.45) holds for all α-admissible $\left\{\phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right\}$, for $i = 1, \ldots, N$, with $\epsilon_{x_0,\alpha,\beta} = \max_i\{x(0)^\top P_{i,\epsilon}x(0)\}$. Finally, note that the cost incurred by player i if the players play the $\epsilon_{\alpha,\beta}$-Nash equilibrium strategies $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$ is

$$\begin{aligned} J_i^* = J_i(x(0), \phi_1^*(x(k)), \ldots, \phi_N^*(x(k)) &= \sum_{k=0}^{\infty} x(k)^\top \left(Q_i + \sum_{j=1}^{N} K_j^{*\top} R_{ij} K_j^*\right) x(k) \\ &= x(0)^\top \left(\sum_{k=0}^{\infty} {A_{cl}^*}^{k\top} \left(Q_i + \sum_{j=1}^{N} K_j^{*\top} R_{ij} K_j^*\right) {A_{cl}^*}^{k}\right) x(0) \\ &= x(0)^\top W_i^* x(0), \end{aligned}$$

where $W_i^* = P_i^* - \Delta W_i$ with ΔW_i as defined in (3.48), which follows from the definition of the controllability Gramian. □

Theorem 3.3.1 provides a way to determine equilibrium strategies for LQ dynamic games, which involves the solution of the coupled inequalities (3.46), $i = 1, \ldots, N$, rather than the equations (3.5), $i = 1, \ldots, N$, and is hence computationally simpler. However, this comes at the cost of introducing a degree of approximation, which is influenced by three main factors:

α : The parameter is a design choice and is related to the convergence rate of the trajectories of system (3.1) in closed loop with the strategies $\left\{\phi_1^*(x(k)), \ldots, \phi_{i-1}^*(x(k)), \hat{\phi}_i(x(k)), \phi_{i+1}^*(x(k)), \ldots, \phi_N^*(x(k))\right\}$ to which an approximate Nash equilibrium solution $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$ is compared in (3.45), for $i = 1, \ldots, N$. The final term on the right-hand side in (3.45) can be rendered arbitrarily small by choosing α such that the closed-loop system (3.52) converges arbitrarily fast. However, the choice of α influences and might increase the value of the term $J_i\left(x(0), \phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right)$ in (3.45). Hence, the choice of α minimising the right-hand side of (3.45) constitutes a trade-off.

β : In practice, this parameter can be selected to ensure that the trajectories of system (3.1) in closed loop with an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$ have a desired "convergence rate" faster than $\frac{1}{\beta}$. This directly influences $\tilde{\Upsilon}_i = \Upsilon_i + P_i^* - \frac{1}{\beta^2}P_i^*$. Hence, the larger $\beta \geq 1$, the larger the additional cost term in (3.51) and hence $\epsilon_{x_0,\alpha,\beta}$.

Υ_i: Similarly, Υ_i, which quantifies how "far" the inequality (3.46) is from holding with equality, affects the additional cost term in (3.51) and hence $\epsilon_{x_0,\alpha,\beta}$. The "magnitude" of Υ_i can be influenced by introducing a constraint of the form (3.44c) for some $\gamma > 0$, such that the feasibility problem to be solved consists in the matrix inequalities (3.44c), (3.46), for $i = 1, \ldots, N$, and the equality (3.44b). By using (3.44c) and (3.49) it can be shown that $\gamma I_n \geq \sum_{i=1}^{N} \Upsilon_i \geq \Upsilon_i$, for $i = 1, \ldots, N$.

Remark 3.3.1. Note that using the result of Theorem 3.3.1 the degree of approximation $\epsilon_{x_0,\alpha,\beta}$ in (3.45) cannot be calculated *a priori*, since it depends on the solution to (3.46), $i = 1, \ldots, N$, and (3.44b), in addition to the initial condition x_0 and the constants α and β relating to the speed of convergence.

To obtain $\epsilon_{\alpha,\beta}$-Nash equilibria of the game (3.1), (3.2), $i = 1, \ldots, N$, with a bound on $\epsilon_{x_0,\alpha,\beta}$, which can be quantified *a priori*, a modification of Theorem 3.3.1 inspired by [62, Theorem 2] is introduced.

Proposition 3.3.1. *Let the conditions stated in Theorem 3.3.1 hold and consider the initial condition* $x(0) = x_0$*. If there exist matrices* K_i^**,* P_i^**, for* $i = 1, \ldots, N$*, satisfying* (3.46)*, for* $i = 1, \ldots, N$*, and* (3.44b)*,* (3.44c) *with* $\gamma > 0$*,* $K_i = K_i^*$ *and* $P_i = P_i^*$*, then the set of strategies* $\{\phi_1^*(x(k)), \ldots, \phi_N^*(x(k))\}$*, with* $\phi_i^*(x(k))$ *as defined in* (3.47)*, for* $i = 1, \ldots, N$*, constitutes an* $\epsilon_{\alpha,\beta}$*-Nash equilibrium of the game* (3.1)*,* (3.2)*,* $i = 1, \ldots, N$*, with*

$$\epsilon_{x_0,\alpha,\beta} = \max_i \left\{ \sum_{k=0}^{\infty} \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k) \right\} \leq \frac{\alpha^2}{\alpha^2 - 1} \gamma \|x_0\|^2. \tag{3.54}$$

Proof. By Theorem 3.3.1, K_i^*, P_i^*, $i = 1, \ldots, N$, satisfying (3.46), for $i = 1, \ldots, N$, and (3.44b) characterise an $\epsilon_{\alpha,\beta}$-Nash equilibrium of the game (3.1), (3.2), $i = 1, \ldots, N$. To obtain the bound on $\epsilon_{x_0,\alpha,\beta}$ in (3.54), recall that $\hat{x}(k)$ denotes the state of system (3.1) in closed loop with any α-admissible set of strategies $\left\{\phi_1^*, \ldots, \phi_{i-1}^*, \hat{\phi}_i, \phi_{i+1}^*, \ldots, \phi_N^*\right\}$, for $i = 1, \ldots, N$, namely (3.52). Using sub-multiplicativity of the matrix norm gives

$$\sum_{k=0}^{\infty} \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k) = \sum_{k=0}^{\infty} \left\| \hat{x}(k)^\top \tilde{\Upsilon}_i \hat{x}(k) \right\| \leq \sum_{k=0}^{\infty} \|\tilde{\Upsilon}_i\| \|\hat{x}(k)\|^2 = \|\tilde{\Upsilon}_i\| \sum_{k=0}^{\infty} \hat{x}(k)^\top \hat{x}(k). \tag{3.55}$$

Note that

$$\sum_{k=0}^{\infty} \hat{x}(k)^\top \hat{x}(k) = x_0^\top \left(\sum_{k=0}^{\infty} \hat{A}_{cl}^{k\,\top} \hat{A}_{cl}^k \right) x_0 = x_0^\top X x_0,$$

where $X = \sum_{k=0}^{\infty} \hat{A}_{cl}^{k\,\top} \hat{A}_{cl}^k$ satisfies the Lyapunov equation $X - \hat{A}_{cl}^\top X \hat{A}_{cl} = I_n$. By α-admissibility, $\rho(\alpha \hat{A}_{cl}) < 1$, and hence it can be shown that

$$I_n \preceq X \preceq \left(\frac{\alpha^2}{\alpha^2 - 1} \right) I_n, \tag{3.56}$$

see e.g. [175]. Finally, it follows from (3.46) and $\tilde{\Upsilon}_i = \Upsilon_i + P_i^* - \frac{1}{\beta^2} P_i^*$ that $\tilde{\Upsilon}_i \preceq P_i^*$. Hence, by (3.44c)

$$\max_i \left(\|\tilde{\Upsilon}_i\| \right) < \gamma. \tag{3.57}$$

Combining (3.55), (3.56) and (3.57) gives (3.54). □

Remark 3.3.2. The result of Proposition 3.3.1 provides conditions characterising $\epsilon_{\alpha,\beta}$-Nash equilibrium solutions with the bound (3.54) on the degree of approximation. Depending only on the initial condition x_0 and the parameters α and γ, which are design choices, (3.54) can be evaluated *a priori.* In addition to using sub-multiplicativity of the induced matrix norm, the bound is derived using α-admissibility to upper bound the solution of a Lyapunov equation and the additional constraint (3.44c), which ensures an upper bound on the solutions of the coupled matrix conditions (3.46), for $i = 1, \dots, N$, and (3.44b). The former is inherently conservative if the value of α is close to 1. The latter can be made less conservative by replacing the constraint (3.44c) with constraints $\gamma I_n - P_i^* \succeq 0$, for $i = 1, \dots, N$, which ensure that $\tilde{\Upsilon}_i \preceq \gamma I_n$, for $i = 1, \dots, N$. Moreover, for either constraint option a solution minimising γ can be sought. However, this makes γ a decision variable and hence comes at the cost of (3.54) no longer being quantifiable *a priori.*

In the following result sufficient conditions are provided for the existence of an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution.

Proposition 3.3.2. *Consider the game* (3.1). (3.2), $i = 1, \dots, N$, *and constant* $\beta \geq 1$. *Assume the pair* $\left(A, \sum_{i=1}^N Q_i\right)$ *is detectable. If the pair* $\left(A, \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix}\right)$ *is* β*-stabilisable, i.e. there exist gains* $\{K_1, \dots, K_N\}$ *such that* $\rho(\beta A_{cl}) < 1$, *and the cost parameters are such that*

$$\begin{bmatrix} \sum_{j=1}^{N}(R_{j1}) - R_{i1} & & \\ & \ddots & \\ & & \sum_{j=1}^{N}(R_{jN}) - R_{iN} \end{bmatrix} \succeq \begin{bmatrix} \sum_{j=1}^{N}(R_{j1}) - R_{11} & & \\ & \ddots & \\ & & \sum_{j=1}^{N}(R_{jN}) - R_{NN} \end{bmatrix} \tag{3.58}$$

for $i = 1, \dots, N$, *then there exists an* $\epsilon_{\alpha,\beta}$*-Nash equilibrium solution of the dynamic game, for any* $\alpha > 1$.

Proof. By detectability of $\left(A, \sum_{i=1}^N Q_i\right)$ and β-stabilisability of $\left(A, \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix}\right)$ there exist (see e.g. [52]) unique matrices $\bar{P} = \bar{P}^\top \succeq 0$ and $\left\{\bar{K}_1, \dots, \bar{K}_N\right\}$ satisfying

$$\bar{P} - (\beta \bar{A}_{cl})^\top \bar{P} (\beta \bar{A}_{cl}) - \sum_{i=1}^{N} Q_i - \sum_{j=1}^{N} \bar{K}_j^\top \left(\sum_{i=1}^{N} R_{ij}\right) \bar{K}_j = 0,$$

$$\begin{bmatrix} \bar{K}_1 \\ \vdots \\ \bar{K}_N \end{bmatrix} = -\left(\begin{bmatrix} \sum_{i=1}^N R_{i1} & & \\ & \ddots & \\ & & \sum_{i=1}^N R_{iN} \end{bmatrix} + \beta^2 B^\top \bar{P} B\right)^{-1} \beta^2 B^\top \bar{P} A,$$

where $B = \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix}$ and $\bar{A}_{cl} = A + \sum_{i=1}^{N} B_i \bar{K}_i$ is β-admissible. Consider instead

$$\begin{bmatrix} K_1' \\ \vdots \\ K_N' \end{bmatrix} = -\left(\begin{bmatrix} R_{11} & & \\ & \ddots & \\ & & R_{NN} \end{bmatrix} + \beta^2 B^\top \bar{P} B \right)^{-1} \beta^2 B^\top \bar{P} A,$$

and let $A_{cl}' = A + \sum_{i=1}^{N} B_i K_i'$. Noting that $\sum_{i=1}^{N} Q_i \succeq Q_i$, if (3.58) holds then it follows from the relation between $\{\bar{K}_1, \dots, \bar{K}_N\}$ and $\{K_1', \dots, K_N'\}$ that

$$\bar{P} - (\beta A_{cl}')^\top \bar{P} (\beta A_{cl}') - Q_i - \sum_{j=1}^{N} {K_j'}^\top R_{ij} K_j' \succeq 0. \tag{3.59}$$

Letting $\bar{P} = \frac{1}{\beta^2} P_i^*$, $i = 1, \dots, N$, and $K_j' = K_j^*$, $j = 1, \dots, N$, (3.44b) holds with $K_i = K_i^*$ and $P_i = P_i^*$ and (3.59) implies (3.46), for $i = 1, \dots, N$. By Theorem 3.3.1, P_i^*, K_i^*, $i = 1, \dots, N$, correspond to an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution of the game. □

Remark 3.3.3. While the condition (3.58) may seem restrictive, it includes the commonly considered case in which $R_{ij} = 0$, for $i = 1, \dots, N$, $j = 1, \dots, N$, $j \neq i$. Recall also that the conditions of Proposition 3.3.2 are sufficient but not necessary, and $\epsilon_{\alpha,\beta}$-Nash equilibria may exist even if (3.58) is violated.

3.3.4 Computation of $\epsilon_{\alpha,\beta}$-Nash equilibria

The results of Theorem 3.3.1 and Proposition 3.3.1 present methods to determine approximate Nash equilibrium solutions by solving feasibility problems involving nonlinear matrix inequality and equality constraints. In [62] it has been shown that similar problems for the continuous-time case, i.e. infinite-horizon LQ differential games, can be reformulated as *bilinear* or *rank constrained* feasibility problems, which despite generally being nondeterministic polynomial-time (NP)-hard are frequently encountered in control theory and have hence been extensively studied (see e.g. [176]). Analogous steps can be used to reformulate the discrete-time problem considered herein[6]. To this end consider the following assumptions.

Assumption 3.3.1. The conditions characterising $\epsilon_{\alpha,\beta}$-Nash equilibria, (3.46), $i = 1, \dots, N$, and (3.44b), admit solutions K_i^*, $P_i^* = {P_i^*}^\top$, such that $P_i^* \succ 0$, for $i = 1, \dots, N$.

Assumption 3.3.2. The cost functional (3.2) of player i is such that either $R_{ij} \succ 0$ or $R_{ij} = 0$ for $i = 1, \dots, N$, $j = 1, \dots, N$, $i \neq j$.

[6]Note that the resulting problem is more complex than the equivalent in the continuous-time case, due to the additional product terms of the decision variables arising in the discrete-time case (see Remark 3.1.2).

If Assumptions 3.3.1 and 3.3.2 hold, then via the Schur complement and a change of variables (3.46) can be converted to the LMI

$$\begin{bmatrix} \frac{1}{\beta^2}P_i^\star & A^\top P_i^* + Y_i^{*\top} & K_i^{\emptyset\top} \\ P_i^* A + Y_i^* & P_i^* & 0 \\ K_i^\emptyset & 0 & R_i^\emptyset \end{bmatrix} \succeq 0, \tag{3.60}$$

for $i = 1, \dots, N$, where $R_i^\emptyset$ is the block diagonal matrix containing all R_{il}, for $l \in \ell$ where $\ell = \{l \mid 1 \leq l \leq N, \text{ and } R_{il} \neq 0\}$ along the main diagonal and $K_i^\emptyset$ is the row wise stacking of the corresponding K_l^*, such that $K_i^{\emptyset\top} R_i^\emptyset K_i^\emptyset = \sum_{l\in\ell} K_l^{*\top} R_{il} K_l^* = \sum_{j=1}^N K_j^{*\top} R_{ij} K_j^*$. A single block row of (3.44b) with $K_i = K_i^*$ and $P_i = P_i^*$ becomes

$$R_{ii}K_i^* + B_i^\top P_i^* A + B_i^\top Y_i^* = 0, \tag{3.61}$$

for $i = 1, \dots, N$, where

$$Y_i^* = P_i^* \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix} \begin{bmatrix} K_1^* \\ \vdots \\ K_N^* \end{bmatrix}, \tag{3.62}$$

for $i = 1, \dots, N$. The conditions (3.60), (3.61) and (3.62), $i = 1, \dots, N$ present an alternative formulation of the conditions characterising $\epsilon_{\alpha,\beta}$-Nash equilibria in Theorem 3.3.1. Determining K_i^*, P_i^* satisfying (3.60), (3.61) and (3.62), $i = 1, \dots, N$, constitutes a bilinear feasibility problem. While efficient solvers for this class of problem exist, depending on the problem it may be desirable from a computational perspective to instead convert the bilinear constraint (3.62) to a rank constraint and an LMI. Note that via the semi-definite embedding lemma [177] and via analogous steps as in [176, Propositon 2] it follows that (3.62) holds if and only if there exist $\mathcal{Y}_i = \mathcal{Y}_i^\top \in \mathbb{R}^{2n\times 2n}$, $\mathcal{Z}_i = \mathcal{Z}_i^\top \in \mathbb{R}^{2n\times 2n}$ such that

$$\operatorname{rank}\left(\begin{bmatrix} \mathcal{Y}_i & 0 \\ 0 & \mathcal{Z}_i \end{bmatrix}\right) \leq 2n, \tag{3.63}$$

and

$$\begin{bmatrix} \mathcal{Y}_i & \begin{bmatrix} \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix} \begin{bmatrix} K_1^* \\ \vdots \\ K_N^* \end{bmatrix} & I_n \\ Y_i^* & P_i^* \end{bmatrix} \\ \begin{bmatrix} \begin{bmatrix} B_1 & \dots & B_N \end{bmatrix} \begin{bmatrix} K_1^* \\ \vdots \\ K_N^* \end{bmatrix} & I_n \\ Y_i^* & P_i^* \end{bmatrix}^\top & \mathcal{Z}_i \end{bmatrix} \succeq 0, \tag{3.64}$$

for $i = 1, \ldots, N$. Determining K_i^*, P_i^* satisfying (3.60), (3.61), (3.63) and (3.64), $i = 1, \ldots, N$, constitutes a rank constraint feasibility problem. This represents another alternative formulation of the conditions characterising $\epsilon_{\alpha,\beta}$-Nash equilibria in Theorem 3.3.1.

Remark 3.3.4. Recall Lemma 3.3.1 and note that using analogous steps as presented above the non-linear SDP (3.44), whose stabilising solutions characterise exact feedback Nash equilibria, can be converted to an optimisation problem involving LMI and linear equality constraints as well as either bilinear or rank constraints.

3.3.5 Example

To illustrate the results presented in this section, consider the problem of macroeconomic policy design for a monetary union consisting of two countries and a common central bank [30, Example 8.15]. The difference in prices between the two countries x satisfies

$$\dot{x} = -x - u_1 + u_2 + 0.5u_3, \tag{3.65}$$

where u_i, $i = 1, 2$, are the national fiscal deficits of the two countries and u_3 is the common interest rate fixed by the central bank. Discretising (3.65) using zero-order hold with time step $\Delta = 0.1$ results in a system of the form (3.1) with $n = 1$, $N = 3$ and $m_i = 1$, for $i = 1, 2, 3$. Let each player seek to minimise a cost functional of the form (3.2), with $Q_1 = 0.1$, $R_{11} = 0.05$, $Q_2 = 0.1$, $R_{22} = 0.1$, $Q_3 = 0.05$, $R_{33} = 0.15$, and $R_{ij} = 0$ for $i = 1, 2, 3$, $j = 1, 2, 3$, $j \neq i$, as illustrated in Figure 3.7. This can be interpreted as country 1 aiming to stabilise the price difference, country 2 being indifferent and the central bank aiming to bring the interest rate to its equilibrium value, which is assumed to be zero. The problem constitutes an LQ dynamic game. For the given system and cost parameters, there exists a unique linear state-feedback Nash equilibrium, which is characterised by the parameters $K_i^\star$, $P_i^\star$, $i = 1, 2, 3$, given in Table 3.3, which satisfy (3.5), $i = 1, 2, 3$ and (3.6). Using the result of Theorem 3.3.1, an $\epsilon_{\alpha,\beta}$-Nash equilibrium solution with $\beta = 1.5$ is charcterised by K_i^*, P_i^*, $i = 1, 2, 3$, given in Table 3.3, which satisfy (3.46) and (3.44b), as well as (3.44c) with $\gamma = 6$. The corresponding $\epsilon_{\alpha,\beta}$-Nash equilibrium strategies satisfy (3.45) with $\epsilon_{x_0,\alpha,\beta} = 0.9175$ for $\alpha = 1.05$. The time histories of the state and inputs with $x_0 = 1$ are shown in Figures 3.8 and 3.9, respectively. The system in closed-loop with the $\epsilon_{\alpha,\beta}$-Nash equilibrium strategies converges faster than the system in closed-loop with the Nash equilibrium strategies. This results in a larger control effort, particularly for country 1, in Figure 3.9. The corresponding Nash equilibrium and $\epsilon_{\alpha,\beta}$-Nash equilibrium costs for each player are given in Table 3.3.

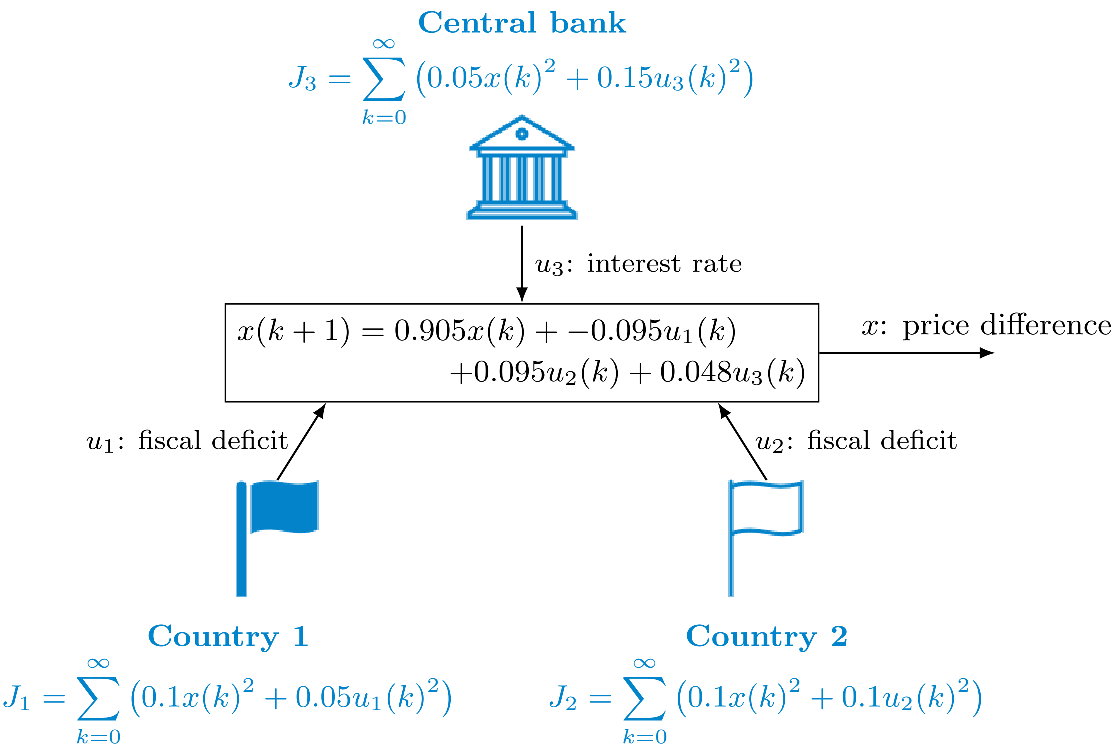


Figure 3.7: Illustration of macroeconomic policy design example.

Table 3.3: Parameters characterising the Nash equilibrium and $\epsilon_{\alpha,\beta}$-Nash equilibrium rounded to four decimal places.

Player	$K_i^\star$	$P_i^\star$	$J_i^\star$	K_i^*	P_i^*	J_i^*
$i=1$	0.5674	0.3618	0.3618	4.2160	4.6311	1.2820
$i=2$	-0.2610	0.3328	0.3328	-0.2448	0.5379	0.1374
$i=3$	-0.0409	0.1566	0.1566	-0.0424	0.2795	0.0652

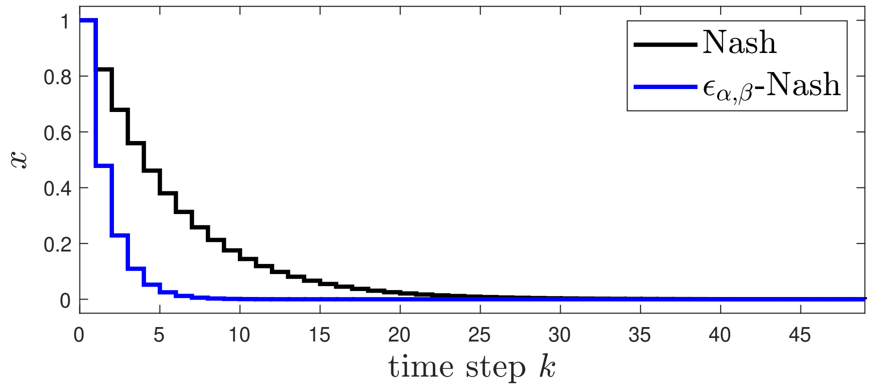


Figure 3.8: Time histories of the states of the closed-loop system with the Nash equilibrium strategies and the $\epsilon_{\alpha,\beta}$-Nash equilibrium strategies.

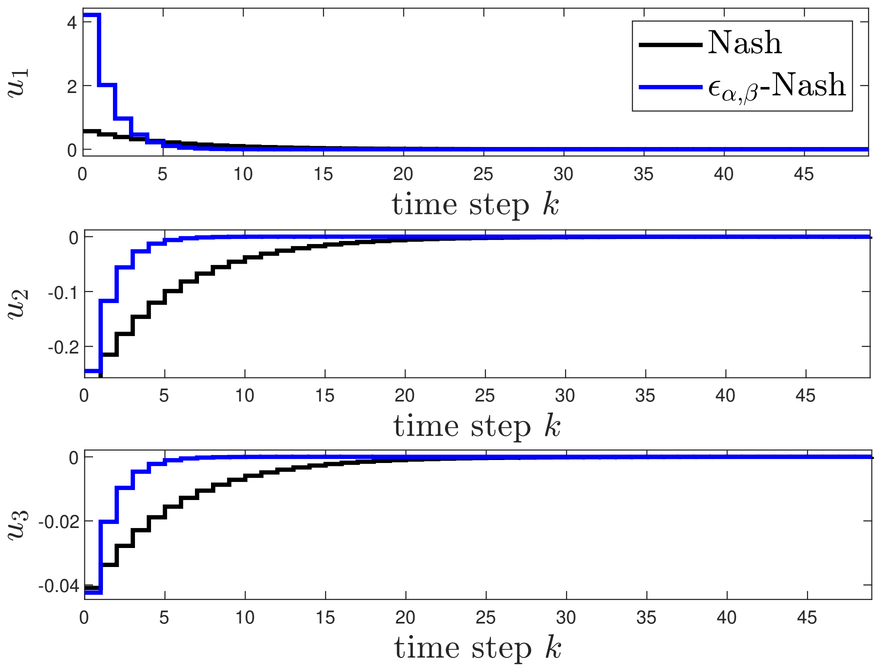


Figure 3.9: Time histories of the Nash equilibrium strategies and the $\epsilon_{\alpha,\beta}$-Nash equilibrium strategies.

3.4 Iterative Nash equilibrium finding algorithms

As highlighted in the previous sections of this chapter, solving the coupled algebraic equations characterising feedback Nash equilibrium solutions of infinite-horizon nonzero-sum LQ dynamic games is generally challenging. In Section 3.3 this has been addressed by considering a weaker notion of solutions, i.e. approximate Nash equilibria. While these are characterised by conditions which are generally easier to solve than the conditions characterising exact Nash equilibria, the resulting feasibility problems are still complex. An alternative approach is to solve the coupled equations numerically. While the literature on iterative solution methods for LQ dynamic games focuses mainly on the continuous-time setting [65, 30, 66, 67, 68, 69], methods in the discrete-time setting include [70] for finite-horizon games and [74, 73, 72] in the context of reinforcement learning. Both in continuous- and discrete-time, most algorithms are presented either without a proof of convergence, or with convergence guarantees limited to the special case in which there exists a unique feedback Nash equilibrium or to an approximate Nash equilibrium.

Focusing on infinite-horizon LQ discrete-time dynamic games, four algorithms are proposed in this section to find a feedback Nash equilibrium solution of a game. The presented iterative schemes rely on solutions of *uncoupled* either Lyapunov or Riccati equations to update the strategy of each player. The first two can be interpreted as policy iteration algorithms and the latter two as value iteration algorithms. Criteria for local convergence of the algorithms to a Nash equilibrium solution are discussed. More precisely, conditions are provided under which a set of Nash equilibrium strategies of the LQ dynamic game, which may admit several of such equilibrium solutions with different outcomes, constitutes a locally asymptotically stable (LAS) equilibrium of the iterative schemes.

The remainder of this section is organised as follows. In Section 3.4.1 the considered problem is introduced. Four iterative algorithms to find Nash equilibria are proposed in Section 3.4.2. The algorithms are discussed in Section 3.4.3. In Section 3.4.4 their performance is demonstrated and discussed via two illustrative numerical examples.

3.4.1 Problem formulation

Consider the LQ infinite-horizon discrete-time dynamic game (3.1), (3.2), $i = 1, \dots, N$, and the problem of determining feedback Nash equilibrium solutions, i.e. admissible[7] feedback strategies of the form $\phi_i(x(k)) = K_i x(k)$ for the players' actions u_i, $i = 1, \dots, N$, such that (2.8) holds (see Definition 2.1.2). In Section 3.1 it is shown that feedback Nash equilibrium solutions for this class of games are characterised by the stabilising solutions of a set of coupled algebraic equations, namely (3.5), $i = 1, \dots, N$, and (3.6), see Theorem 3.1.1. Typically, determining $K_i^\star$ and $P_i^\star$, for $i = 1, \dots, N$, which satisfy (3.5), $i = 1, \dots, N$, and (3.6), and thereby determining feedback Nash equilibrium

[7] In the sense of Definition 3.3.1.

solutions of the game (3.1), (3.2), $i = 1, \dots, N$, is not a straightforward task. Hence, the problem of finding a feedback Nash equilibrium solution of the game iteratively, by solving matrix equations of reduced complexity with respect to (3.5), $i = 1, \dots, N$, and (3.6) at each iteration step, is considered in this section. This is formalised in the following statement.

Problem 3.4.1. Given an initial guess $K_i^{(0)}$, for $i = 1, \dots, N$, iteratively determine gains $K_i^\star$, for $i = 1, \dots, N$, such that the corresponding set of feedback strategies $\{\phi_1^\star(x(k)), \dots, \phi_N^\star(x(k))\}$, with $\phi_i^\star(x(k))$ given by (3.3), for $i = 1, \dots, N$, constitutes a feedback Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \dots, N$.

In the remainder of this section, iterative algorithms addressing Problem 3.4.1 are proposed and analysed.

3.4.2 Iterative algorithms for Nash equilibria

Considering the game (3.1), (3.2), $i = 1, \dots, N$, and Problem 3.4.1, four methods are proposed to determine a solution $K_i^\star$, $P_i^\star$ to (3.5), $i = 1, \dots, N$, and (3.6), starting from an initial guess $K_i^{(0)}$, by iteratively updating the solution guesses $K_i^{(l)}$, $P_i^{(l)}$, for $i = 1, \dots, N$ and $l \in \mathbb{N}$. The first two algorithms involve update laws based on the solution of Lyapunov equations, and are hence referred to as "Lyapunov iterations" inspired by the name for a related method for the continuous-time setting [66]. The latter two algorithms are based on the solution of Riccati equations and are hence referred to as "Riccati iterations". Each type of algorithm is proposed in a *synchronous* as well as in an *asynchronous* fashion, as detailed below and as illustrated in Figure 3.10.

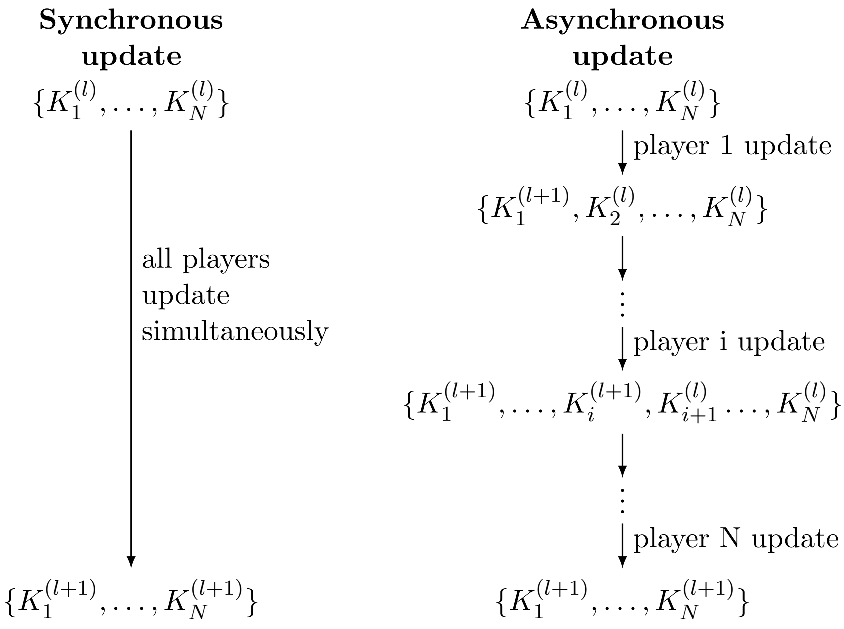


Figure 3.10: Illustration of the synchronous and asynchronous strategy update types.

Synchronous updates: The strategy update of player i at iteration $(l+1)$ is defined on the premise that the actions of all other players remain fixed at feedback strategies with the gains corresponding to the previous iteration step (l), i.e. $u_j(k) = K_j^{(l)} x(k)$, for $j = 1, \ldots, N$, $j \neq i$. Hence, the system dynamics perceived by player i at iteration (l) are

$$x(k+1) = \hat{A}_{s,i}^{(l)} x(k) + B_i u_i(k),$$

with $\hat{A}_{s,i}^{(l)} = A + \sum_{j=1, j\neq i}^{N} B_j K_j^{(l-1)}$, and the cost functional of player i becomes

$$J_i\left(x(0), u_i(\cdot)\right) = \sum_{k=0}^{\infty} \left(x(k)^\top \hat{Q}_{s,i}^{(l)} x(k) + u_i(k)^\top R_{ii} u_i(k) \right),$$

with $\hat{Q}_{s,i}^{(l)} = Q_i + \sum_{j=1, j\neq i}^{N} K_j^{(l-1)^\top} R_{ij} K_j^{(l-1)}$.

Asynchronous updates: The strategy update of player i at iteration $(l+1)$ takes into account the updates of players $w = 1, \ldots, i-1$ at the same iteration. Hence, player i treats the strategies of players w fixed at $u_w(k) = K_w^{(l+1)} x(k)$, for $w = 1, \ldots, i-1$ and players j fixed at $u_j(k) = K_j^{(l)} x(k)$, for $j = i+1, \ldots, N$. The system dynamics perceived by player i at iteration (l) are thus

$$x(k+1) = \hat{A}_{a,i}^{(l)} x(k) + B_i u_i(k),$$

with $\hat{A}_{a,i}^{(l)} = A + \sum_{w=1}^{i-1} B_w K_w^{(l)} + \sum_{j=i+1}^{N} B_j K_j^{(l-1)}$, and the cost functional of player i becomes

$$J_i\left(x(0), u_i(\cdot)\right) = \sum_{k=0}^{\infty} \left(x(k)^\top \hat{Q}_{a,i}^{(l)} x(k) + u_i(k)^\top R_{ii} u_i(k) \right),$$

with $\hat{Q}_{a,i}^{(l)} = Q_i + \sum_{w=1}^{i-1} K_w^{(l)^\top} R_{iw} K_w^{(l)} + \sum_{j=i+1}^{N} K_j^{(l-1)^\top} R_{ij} K_j^{(l-1)}$.

To streamline the presentation, let $\hat{A}_{\sigma,i}^{(l)}$ and $\hat{Q}_{\sigma,i}^{(l)}$ denote the dynamics matrix and state cost weight, respectively, associated with a generic update rule, where $\sigma = s$ denotes the synchronous update and $\sigma = a$ denotes the asynchronous update. Moreover, to streamline the convergence analysis of the algorithms presented below, consider a generic equilibrium finding algorithm. More precisely, let $z^\star \in \mathbb{R}^p$ satisfy the set of algebraic equations

$$0 = L(z^\star), \tag{3.66}$$

for some $L : \mathbb{R}^p \to \mathbb{R}^p$ and consider an iterative update law for the solution guess $z^{(l)}$, $l \in \mathbb{N}$, satisfying the implicit relation

$$0 = F\left(z^{(l+1)}, z^{(l)}\right), \tag{3.67}$$

with $F : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}^p$ differentiable and such that $0 = F(z^\star, z^\star)$. The first order approximation of (3.67) around the equilibrium $z^\star$ is given by

$$0 = M_n\left(z^{(l+1)} - z^\star\right) + M_c\left(z^{(l)} - z^\star\right), \tag{3.68}$$

where

$$M_n = \left.\frac{\partial F\left(z^{(l+1)}, z^{(l)}\right)}{\partial z^{(l+1)}}\right|_{z^\star, z^\star}, \tag{3.69a}$$

$$M_c = \left.\frac{\partial F\left(z^{(l+1)}, z^{(l)}\right)}{\partial z^{(l)}}\right|_{z^\star, z^\star}. \tag{3.69b}$$

Letting $\delta z^{(l)} = \left(z^{(l)} - z^\star\right)$ the dynamics of the iterative update law satisfying (3.67) can be described by

$$\delta z^{(l+1)} = H\delta z^{(l)}, \tag{3.70}$$

in a neighbourhood of $z^\star$, with $H = -M_n^{-1}M_c$, assuming the matrix M_n is invertible.

Lemma 3.4.1. *If H in* (3.70) *is Schur, i.e. $\rho(H) < 1$, and $z^{(0)}$ lies in a neighbourhood of $z^\star$, the iterative algorithm with update law satisfying* (3.67) *asymptotically converges to the equilibrium $z^\star$ of* (3.66).

Proof. The claim follows from Lyapunov's indirect method. □

Remark 3.4.1. Lemma 3.4.1 provides conditions ensuring that a solution $z^\star$ of the set of algebraic equations (3.66) is a locally stable fixed point of the nonlinear recurrence relation (3.67). In the following, Lemma 3.4.1 is employed to analyse whether a feedback Nash equilibrium of the game (3.1), (3.2), $i = 1, \ldots, N$, which is characterised by the set of feedback gains $\{K_1^\star, \ldots, K_N^\star\}$ satisfying the algebraic equations (3.5), $i = 1, \ldots, N$, and (3.6), for some $\{P_1^\star, \ldots, P_N^\star\}$, is attractive for the iterative update laws proposed below, which are nonlinear recurrence relations, or in other words, nonlinear discrete-time dynamical systems (see e.g. [178]). This local convergence analysis approach is chosen because if there are multiple feedback Nash equilibria for a given game, it is inherent that the convergence properties of any algorithm to a specific equilibrium can only be local. The results of Section 3.2 highlight that even a simple two-player game involving scalar dynamics may admit multiple feedback Nash equilibria. Naturally, it is then of interest to characterise the set of initial conditions for which an iterative method converges to a certain solution. Determining the basin of attraction of a stable fixed point of a nonlinear dynamical system is generally challenging, and approximations are commonly sought numerically. An estimate of the basin of attraction can be found via La Salle's local invariant set theorem (see e.g. [179]). The challenge in this context is finding a suitable Lyapunov function. The common and straightforward choice of a quadratic Lyapunov function based on the linearisation of the nonlinear system around the fixed point often results in a conservative estimate of the region of attraction, especially for systems with many states, since the approach yields an ellipsoid in state space, whereas the basin of attraction may have a very different shape [180]. There exists a rich literature in this context in particular for continuous-time systems, see e.g. [181, 180, 182] and references therein. For discrete-time systems, additional challenges in the analysis of the state space and characterisation of stability regions arise. This is due to trajectories being a sequence of isolated points rather than curves in state space and the fact that backward trajectories may not be defined or may not be unique [183]. As a result, a basin of attraction may not be a connected or invariant set [184]. Approaches to estimate basins of attraction of fixed points of nonlinear discrete-time systems can be found e.g. in [184, 185, 186, 187, 188].

To use Lemma 3.4.1 to analyse convergence of the algorithms, consider the following result introducing conditions related to the solution of the set of coupled matrix equations (3.5), $i = 1, \dots, N$, and (3.6).

Lemma 3.4.2. *Let the set of feedback gains* $\{K_1^\star, \dots, K_N^\star\}$*, satisfying* (3.4), (3.5), $i = 1, \dots, N$*, and* (3.6) *for some* $\{P_1^\star, \dots, P_N^\star\}$*, correspond to any feedback Nash equilibrium solution of the game* (3.1), (3.2), $i = 1, \dots, N$*. Let* $z^\star = \begin{bmatrix} \mathrm{vec}(K_1^\star)^\top & \dots & \mathrm{vec}(K_N^\star)^\top \end{bmatrix}^\top$*. The vector* $z^\star$ *is such that* $L(z^\star) = 0$*, with* $L(z^\star) = \begin{bmatrix} L_1^\top & \dots & L_N^\top \end{bmatrix}^\top$*, and*

$$L_i = \mathrm{vec}\left(R_{ii}K_i^\star\right) - \left((A + \sum_{j=1}^{N} B_j K_j^\star)^\top \otimes B_i^\top\right) \times \left((A + \sum_{j=1}^{N} B_j K_j^\star)^\top \otimes (A + \sum_{j=1}^{N} B_j K_j^\star)^\top - I_{n^2}\right)^{-1} \mathrm{vec}\left(Q_i + \sum_{j=1}^{N} K_j^{\star\top} R_{ij} K_j^\star\right), \quad (3.71)$$

for $i = 1, \dots, N$*. Conversely, any* $\{K_1^\star, \dots, K_N^\star\}$ *satisfying* (3.71), $i = 1, \dots, N$*, and* (3.4) *is a set of feedback gains corresponding to a Nash equilibrium solution of the game* (3.1), (3.2), $i = 1, \dots, N$*.*

Proof. The entries of the mapping L given in (3.71) are derived from the vectorisation of (3.5), as well as the vectorisation of a single block row of (3.6) and by eliminating $\mathrm{vec}\left(P_i^\star\right)$. Hence, any $K_i^\star$, $i = 1, \dots, N$, satisfying (3.5), $i = 1, \dots, N$, and (3.6) satisfies (3.71), $i = 1, \dots, N$. Since (3.4) holds, (3.5) has a unique solution $P_i^\star$, $i = 1, \dots, N$, for fixed $K_i^\star$, $i = 1, \dots, N$. Moreover, vectorisation is a linear transformation. Hence, conversely, any $K_i^\star$, $i = 1, \dots, N$, such that (3.4) holds and satisfying (3.71) satisfies (3.5), $i = 1, \dots, N$, and (3.6) for some $P_i^\star$, for $i = 1, \dots, N$. □

Lyapunov iterations

Consider the following iterative update law, defined with respect to the unified notation for synchronous ($\sigma = s$) and asynchronous ($\sigma = a$) algorithms introduced above,

$$0 = \hat{Q}_{\sigma,i}^{(l+1)} + K_i^{(l)\top} R_{ii} K_i^{(l)} - P_i^{(l+1)} + \left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l)}\right)^\top P_i^{(l+1)} \left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l)}\right), \tag{3.72a}$$

$$K_i^{(l+1)} = -\left(R_{ii} + B_i^\top P_i^{(l+1)} B_i\right)^{-1} B_i^\top P_i^{(l+1)} \hat{A}_{\sigma,i}^{(l+1)}, \tag{3.72b}$$

for $i = 1, \ldots, N$, $l \in \mathbb{N}$. Note that (3.72a) is a *Lyapunov* equation, whereas (3.72b) assigns a value to K_i at each step $(l+1)$. In the following result conditions for local asymptotic convergence of the update law (3.72), for $i = 1, \ldots, N$, to a solution $\{K_1^\star, \ldots, K_N^\star\}$, $\{P_1^\star, \ldots, P_N^\star\}$ of (3.4), (3.5), $i = 1, \ldots, N$, and (3.6) are provided.

Proposition 3.4.1. *Consider Problem 3.4.1 and let the set of feedback gains* $\{K_1^\star, \ldots, K_N^\star\}$*, satisfying* (3.4)*,* (3.5)*,* $i = 1, \ldots, N$*, and* (3.6)*, for some* $\{P_1^\star, \ldots, P_N^\star\}$*, correspond to any feedback Nash equilibrium solution of the game* (3.1)*,* (3.2)*,* $i = 1, \ldots, N$*. Consider the iterative update law* (3.72)*, for* $i = 1, \ldots, N$*. Let* $z^{(l)} = \begin{bmatrix} \mathrm{vec}\left(K_1^{(l)}\right)^\top & \ldots & \mathrm{vec}\left(K_N^{(l)}\right)^\top \end{bmatrix}^\top$ *and* $F(z^{(l)}, z^{(l+1)}) = \begin{bmatrix} F_1^\top & \ldots & F_N^\top \end{bmatrix}^\top$*, with*

$$\begin{aligned} F_i = {} & \mathrm{vec}\left(R_{ii} K_i^{(l+1)}\right) - \left((\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)})^\top \otimes B_i^\top\right) \\ & \times \left((\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l)})^\top \otimes (\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l)})^\top - I_{n^2}\right)^{-1} \mathrm{vec}\left(\hat{Q}_{\sigma,i}^{(l+1)} + K_i^{(l)\top} R_{ii} K_i^{(l)}\right), \end{aligned} \tag{3.73}$$

for $i = 1, ..., N$*. Suppose that*

i. the eigenvalues λ_j*,* $j = 1, \ldots, p$*,* $p \leq n$ *of* $\left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l)}\right)$ *are such that* $\lambda_j \lambda_q \neq 1$ *for all* $j = 1, \ldots, p$*,* $q = 1, \ldots, p$ *and* $i = 1, \ldots, N$*;*

ii. the matrix $H = -M_n^{-1} M_c$ *is Schur, where* M_n *and* M_c *are constructed as in* (3.69) *with respect to* F *with the components* F_i*,* $i = 1, \ldots, N$*, as defined by* (3.73)*.*

Then $\{K_1^\star, \ldots, K_N^\star\}$ *is a LAS equilibrium of the synchronous* ($\sigma = s$) *or asynchronous* ($\sigma = a$) *Lyapunov iterations algorithm* (3.72)*, for* $i = 1, \ldots, N$*.*

Proof. Note that the entries of the mapping F given in (3.73), are derived from the vectorisation of the (matrix) difference equations (3.72), for $i = 1, \ldots, N$, and by eliminating $\mathrm{vec}\left(P_i^{(l+1)}\right)$. Hence, any $K_i^{(l)}$, $K_i^{(l+1)}$, $i = 1, \ldots, N$, satisfying (3.72) satisfy (3.73), for $i = 1, \ldots, N$. Condition *i.* ensures that there exists a unique solution $P_i^{(l+1)}$ to (3.72a) [189]. Hence, since vectorisation is a linear transformation, any $K_i^{(l)}$, $K_i^{(l+1)}$, $i = 1, \ldots, N$, satisfying (3.73) satisfy (3.72), for $i = 1, \ldots, N$. Note that $0 = F(z^{(l+1)}, z^{(l)})$ and $0 = F(z^\star, z^\star)$. With this notation in place, local asymptotic stability follows directly from Lemma 3.4.1 and Lemma 3.4.2. □

Proposition 3.4.1 ensures the existence of a non-empty basin of attraction with respect to the discrete-time nonlinear system (3.72) for all Nash equilibrium strategies which satisfy the stated conditions. Note that the conditions of Proposition 3.4.1 cannot in fact be verified *a priori* since the conditions of item *i.* depend on the strategy updates and the conditions of item *ii.* depend on the knowledge of the Nash equilibrium solution sought for.

Remark 3.4.2. The synchronous version of the Lyapunov iterations (3.72) can be interpreted as a discrete-time, N-player equivalent of the algorithm presented in [66]. Note that due to the additional product terms of the decision variables arising in (3.6) compared to the continuous-time case, different interpretations of the discrete-time synchronous Lyapunov iterations are possible. In an alternative version to the one presented herein, the i-th player updates $P_i^{(l+1)}$ using (3.72a), for $i = 1, \ldots, N$. However, in place of (3.72b),

$$\begin{bmatrix} R_{11} + B_1^\top P_1^{(l+1)} B_1 & \ldots & B_1^\top P_1^{(l+1)} B_N \\ \vdots & \ddots & \vdots \\ B_N^\top P_N^{(l+1)} B_1 & \ldots & R_{NN} + B_N^\top P_N^{(l+1)} B_N \end{bmatrix} \begin{bmatrix} K_1^{(l+1)} \\ \vdots \\ K_N^{(l+1)} \end{bmatrix} = - \begin{bmatrix} B_1^\top P_1^{(l+1)} \\ \vdots \\ B_N^\top P_N^{(l+1)} \end{bmatrix} A. \tag{3.74}$$

is used to compute the update of $K_i^{(l+1)}$ for all i simultaneously. This implies that $P_i^{(l+1)}$ is first computed for all $i = 1, \ldots, N$, and then used to compute $K_i^{(l+1)}$ for $i = 1, \ldots, N$, which is inherently a centralised approach. This version has been presented in [73, Algorithm 1] for the two-player case. Note that in contrast to the version presented herein, namely (3.72) with $\sigma = s$, the scheme in [73, Algorithm 1] requires the invertibility of the matrix

$$\begin{bmatrix} R_{11} + B_1^\top P_1^{(l+1)} B_1 & \ldots & B_1^\top P_1^{(l+1)} B_N \\ \vdots & \ddots & \vdots \\ B_N^\top P_N^{(l+1)} B_1 & \ldots & R_{NN} + B_N^\top P_N^{(l+1)} B_N \end{bmatrix}.$$

Another similar algorithm has been presented in [74, Algorithm 1]. While the update law of this algorithm is equivalent to (3.72) with $\sigma = s$, the algorithm in [74] is also implemented in a centralised fashion, namely, $P_i^{(l+1)}$ is first computed for all $i = 1, \ldots, N$, and then used to compute $K_i^{(l+1)}$ for $i = 1, \ldots, N$. In contrast, in the version presented herein, the i-th player updates $P_i^{(l+1)}$ and $K_i^{(l+1)}$ together by solving (3.72) and the updates of the different players happen sequentially. This choice is motivated by the fact that it allows a data-driven implementation as discussed in Chapter 5 in which each player's strategy is updated in a distributed fashion.

Riccati iterations

Consider the iterative update law given by the stabilising solution of the set of algebraic equations

$$0 = \hat{Q}_{\sigma,i}^{(l+1)} + K_i^{(l+1)^\top} R_{ii} K_i^{(l+1)} - P_i^{(l+1)} + \left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)}\right)^\top P_i^{(l+1)} \left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)}\right), \tag{3.75a}$$

$$K_i^{(l+1)} = -\left(R_{ii} + B_i^\top P_i^{(l+1)} B_i\right)^{-1} B_i^\top P_i^{(l+1)} \hat{A}_{\sigma,i}^{(l+1)}, \tag{3.75b}$$

for $i = 1, \dots, N$, $l \in \mathbb{N}$. As above two update scenarios are discussed: a synchronous one ($\sigma = s$) and an asynchronous one ($\sigma = a$). Note that (3.75a) is a *Riccati* equation, whereas (3.75b) relates $K_i^{(l+1)}$ and $P_i^{(l+1)}$.

Remark 3.4.3. The Riccati iterations (3.75) can be interpreted as player i solving an LQR problem at each iteration step to update the feedback gain K_i, minimising player i's own cost functional (3.2) subject to the dynamics (3.1), with the actions of the other players j, $j = 1, \dots, N$, $j \neq i$, fixed at state-feedback strategies (which do not necessarily correspond to a Nash equilibrium solution of the game), for $i = 1, \dots, N$.

In the following result conditions for local asymptotic convergence of the update law (3.75), for $i = 1, \dots, N$, to a solution $\{K_1^\star, \dots, K_N^\star\}$, $\{P_1^\star, \dots, P_N^\star\}$ of (3.4), (3.5), $i = 1, \dots, N$, and (3.6) are provided.

Proposition 3.4.2. *Consider Problem 3.4.1 and let the set of feedback gains* $\{K_1^\star, \dots, K_N^\star\}$*, satisfying* (3.4)*,* (3.5)*,* $i = 1, \dots, N$ *and* (3.6)*, for some* $\{P_1^\star, \dots, P_N^\star\}$*, correspond to any feedback Nash equilibrium solution of the game* (3.1)*,* (3.2)*,* $i = 1, \dots, N$*. Consider the iterative update law* (3.75)*, for* $i = 1, \dots, N$*. Let* $z^{(l)} = \begin{bmatrix} \mathrm{vec}(K_1^{(l)})^\top & \dots & \mathrm{vec}(K_N^{(l)})^\top \end{bmatrix}^\top$ *and* $F(z^{(l)}, z^{(l+1)}) = \begin{bmatrix} F_1^\top & \dots & F_N^\top \end{bmatrix}^\top$*, with*

$$\begin{aligned} F_i = \mathrm{vec}\left(R_{ii} K_i^{(l+1)}\right) - &\left((\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)})^\top \otimes B_i^\top\right) \\ &\times \left((\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)})^\top \otimes (\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)})^\top - I_{n^2}\right)^{-1} \\ &\times \mathrm{vec}\left(\hat{Q}_{\sigma,i}^{(l+1)} + K_i^{(l+1)^\top} R_{ii} K_i^{(l+1)}\right), \end{aligned} \tag{3.76}$$

for $i = 1, \dots, N$*. If the matrix* $H = -M_n^{-1} M_c$ *is Schur, where* M_n *and* M_c *are constructed as in* (3.69) *with respect to* F *with the components* F_i*,* $i = 1, \dots, N$*, as defined by* (3.76)*, then* $\{K_1^\star, \dots, K_N^\star\}$ *is a LAS equilibrium of the synchronous (*$\sigma = s$*) or asynchronous (*$\sigma = a$*) Riccati iterations algorithm* (3.75)*, for* $i = 1, \dots, N$*.*

Proof. Note that the entries of the mapping F given in (3.76), are derived from the vectorisation of the (matrix) difference equations (3.75), for $i = 1, \ldots, N$, and by eliminating $\text{vec}(P_i^{(l+1)})$. Hence, any $K_i^{(l)}$, $K_i^{(l+1)}$, $i = 1, \ldots, N$, satisfying (3.75) also satisfy (3.76), for $i = 1, \ldots, N$. By construction, the strategy updates are such that the zero equilibrium of $x(k+1) = \left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)}\right) x(k)$ is stable. Thus, for fixed $K_i^{(l)}$, $K_i^{(l+1)}$, $i = 1, \ldots, N$, there exists a unique solution $P_i^{(l+1)}$ to (3.75a). Hence, since vectorisation is a linear transformation, any $K_i^{(l)}$, $K_i^{(l+1)}$, $i = 1, \ldots, N$, satisfying (3.76) satisfy (3.75), for $i = 1, \ldots, N$. Note that $0 = F\left(z^{(l+1)}, z^{(l)}\right)$ and $0 = F\left(z^\star, z^\star\right)$. With this notation in place, local asymptotic stability follows directly from Lemma 3.4.1 and Lemma 3.4.2. □

Proposition 3.4.2 ensures the existence of a non-empty basin of attraction with respect to the discrete-time nonlinear system (3.75) for all Nash equilibrium strategies which satisfy the stated conditions. As in the case of the Lyapunov iterations in Proposition 3.4.1, the conditions which ensure local convergence in Proposition 3.4.2 cannot be verified *a priori* since they depend on the knowledge of the Nash equilibrium solution sought for. Differently from several iterative schemes in the literature (e.g. [66, 73, 72, 63]), the iterations (3.75), $i = 1, \ldots, N$, do not require the initial guess $K_i^{(0)}$, for $i = 1, \ldots, N$, to be stabilising to converge to a stabilising solution of (3.5), $i = 1, \ldots, N$, and (3.6). Note that in the asynchronous case ($\sigma = a$) no initial guess $K_1^{(0)}$ is needed for player 1.

Remark 3.4.4. The asynchronous version of the Riccati iterations (3.75) is the discrete-time and N-player equivalent to [68, Algorithm 4.7]. The result therein is provided without any (*a priori* or *a posteriori*) certificate of convergence.

3.4.3 Discussion

The conditions of Propositions 3.4.1 and 3.4.2 ensure that the iterations (3.72) and (3.75), which involve solving a sequence of *uncoupled* Lyapunov or Riccati equations, respectively, for each player at each iteration, are locally asymptotically convergent to a stabilising solution of the *coupled* matrix equations (3.5), $i = 1, \ldots, N$, and (3.6), which characterises a feedback Nash equilibrium solution. However, no comment has so far been made regarding the *recursive feasibility of the update laws* and *stability properties* of the system (3.1) in closed loop with the current strategy guesses of players $i = 1, \ldots, N$. These properties are particularly relevant if the strategy updates are implemented online, as is the case in the data-driven versions of the algorithms, which are introduced in Chapter 5.

Recursive feasibility: The Lyapunov updates (3.72) are feasible at iteration $(l+1)$ if there exists a unique solution $P_i^{(l+1)}$ to (3.72a). Such a solution exists if (see e.g. [189])

(LI 1) condition *i.* of Proposition 3.4.1 holds.

The Riccati updates (3.75) are feasible at iteration $(l+1)$ if (3.75) admits a unique stabilising solution $K_i^{(l+1)}$, $P_i^{(l+1)} = P_i^{(l+1)\top} \succeq 0$. Such a solution exists (recall that by definition $R_{ii} \succ 0$, $Q_i \succeq 0$, and let $\hat{Q}_{\sigma,i}^{(l+1)} = C_{\sigma,i}^{(l+1)\top} C_{\sigma,i}^{(l+1)}$) if (see e.g. [52])

(RI 1) the pair $\left(\hat{A}_{\sigma,i}^{(l+1)}, B_i\right)$ is stabilisable for $i = 1, \ldots, N$.

(RI 2) the pair $\left(\hat{A}_{\sigma,i}^{(l+1)}, C_{\sigma,i}^{(l+1)}\right)$ is detectable for $i = 1, \ldots, N$.

Note that the condition (RI 2) is always satisfied, for $i = 1, \ldots, N$, if $Q_i \succ 0$, or if the pair (A, Q_i) is observable and $R_{ij} \succ 0$ for all $j = 1, \ldots, N$. These stronger conditions are more easily verified than (RI 2), as they do not depend on the strategy guesses and can be checked *a priori*. In the case of the asynchronous Riccati iterations (3.75), $\sigma = a$, condition (RI 1) is always satisfied if the pair (A, B_i), for $i = 1, \ldots, N$, is stabilisable. Hence, if this is the case and condition (RI 2) holds, then the asynchronous Riccati iterations are guaranteed to be recursively feasible.

Stability: Consider system (3.1) in closed loop with the players' current strategy guesses at iteration $(l + 1)$. In the case of synchronous updates this results in the dynamics $x(k + 1) = \left(A + \sum_{j=1}^{N} B_j K_j^{(l)}\right) x(k) = A_{cl,s}^{(l+1)} x(k)$. In the case of asynchronous updates this results in the dynamics $x(k + 1) = \left(\hat{A}_{a,i}^{(l+1)} + B_i K_i^{(l+1)}\right) x(k) = A_{cl,a,i}^{(l+1)} x(k)$ after the update of player i, for $i = 1, \ldots, N$, since the players update their strategy sequentially, as illustrated in Figure 3.10. Consider first the Lyapunov iterations (3.72). If there exists $P_i^{(l+1)} = P_i^{(l+1)^\top} \succ 0$ satisfying (3.72a), with $\sigma = s$ for any $i = 1, \ldots, N$ and for all $l \in \mathbb{N}$, it follows that the recursive strategy updates obtained via the *synchronous* Lyapunov iterations are stabilising, namely $\rho\left(A_{cl,s}^{(l+1)}\right) < 1$, for all $l \in \mathbb{N}$. Similarly, if there exists $P_i^{(l+1)} = P_i^{(l+1)^\top} \succ 0$ satisfying (3.72a) with $\sigma = a$ for all $i = 1, \ldots, N$, and for all $l \in \mathbb{N}$, it follows that the recursive strategy updates obtained via the *asynchronous* Lyapunov iterations are stabilising, namely $\rho\left(A_{cl,a,i}^{(l+1)}\right) < 1$, for $i = 1, \ldots, N$, for all $l \in \mathbb{N}$.
Consider now the Riccati iterations (3.75). By construction, (3.75) ensures that $\rho\left(\hat{A}_{\sigma,i}^{(l+1)} + B_i K_i^{(l+1)}\right) < 1$ at iteration $(l + 1)$ for player i. However, in the case of *synchronous* updates $(\sigma = s)$ this does not provide any guarantees for $A_{cl,s}^{(l+1)}$. For the asynchronous updates $(\sigma = a)$, on the other hand, this ensures that $\rho\left(A_{cl,a,i}^{(l+1)}\right) < 1$ after the update of player i. Hence, if (3.75) (with $\sigma = a$) is feasible for all $i = 1, \ldots, N$ and for all $l \in \mathbb{N}$, then the recursive strategy updates obtained via the *asynchronous* Riccati iterations are stabilising.

Remark 3.4.5. In practice, it is desirable for the initial guess of the matrices $K_i^{(0)}$, for $i = 1, \ldots, N$, for the Lyapunov iterations (3.72) to be such that the resulting closed-loop system is asymptotically stable to search for a positive semi-definite solution to the Lyapunov equation (3.72a). Albeit involving the solution of more complex algebraic equations at each iteration, the Riccati iterations (3.75) admit general (non-stabilising) initial guesses of the matrices $K_i^{(0)}$, for $i = 1, \ldots, N$.

Remark 3.4.6. In the context of reinforcement learning (see e.g. [52, 190]), the Lyapunov iterations (both synchronous and asynchronous) can be interpreted as *policy iteration* algorithms, with (3.72a) representing the policy evaluation step and (3.72b) the policy update. The (synchronous and asynchronous) Riccati iterations, on the other hand, can be interpreted as *value iteration* algorithms, since the update law involving the solution of a Riccati equation corresponds to minimising the value function for each player characterised by P_i, $i = 1, \ldots, N$, at each update step as noted in Remark 3.4.3. The specific nature of this minimisation problem makes it possible to formulate the update law in terms of the gains K_i, $i = 1, \ldots, N$, characterising the control strategies (policies) analogous to the policy iteration algorithms.

3.4.4 Examples

The efficacy of the presented algorithms is demonstrated via two illustrative examples. The first involves a scalar two-player game. In this relatively simple case all feedback Nash equilibria of the game can be computed analytically and the two-player set-up makes it straightforward to visualise the regions of convergence of the algorithms graphically. This allows to illustrate and compare the performance of the different algorithms. The second example demonstrates the performance of the algorithms for a game involving a slightly larger state-dimension $n = 2$ and $N = 4$ players and facilitates a comparison to alternative algorithms in the literature for the considered class of games.

Scalar two-player example

To illustrate the efficacy of the proposed algorithms, consider the scalar numerical example described by system (3.1), with $N = 2$, $A = 1.0947$, $B_1 = 0.10254$, $B_2 = 0.045934$, and the cost functionals (3.2), $i = 1, 2$, with $Q_1 = 0.11112$, $Q_2 = 0.25806$, $R_{11} = 0.40872$, $R_{22} = 0.5949$, $R_{12} = R_{21} = 0$. For the given system and cost parameters, there exist three stabilising solutions to (3.5), $i = 1, \ldots, N$, and (3.6). The corresponding values of $P_i^\star$ and $K_i^\star$, for $i = 1, 2$, are given in Table 3.4.

The update histories for the four proposed algorithms starting from initial guess $K_1^{(0)} = -1$, $K_2^{(0)} = -2$, are shown in Figure 3.11. All four algorithms converge to NE 3. The Lyapunov iterations converge to $\max_i \left(\left\| K_i^{(l+1)} - K_i^{(l)} \right\| \right) \leq 10^{-5}$ within 15 iterations (synchronous) and 10 iterations (asynchronous), whereas the Riccati iterations take 19 iterations (synchronous) and 11 iterations (asynchronous). Note that in both cases the asynchronous update converges faster than the synchronous update. Figure 3.12 gives an insight into the regions of attraction of the gains corresponding to the three feedback Nash equilibrium solutions. It is worth noting that NE 1 is not attractive for any of the algorithms and that there are significant differences in the regions of attraction for NE 2 and NE 3 between the four algorithms. Apart from a few exceptions, the Lyapunov iterations (3.72) require a stabilising initial guess to converge to a stabilising solution of (3.5), $i = 1, \ldots, N$, and (3.6), whereas the Riccati iterations do not require a stabilising initial guess. In fact, for the considered example and range of initial conditions the asynchronous Riccati iterations always converge to either NE 3 or NE 2, with the regions of attraction separated by the line $K_2 = K_2^\star(\text{NE } 1)$. For the synchronous Riccati iterations on the other hand, an interesting behaviour is observed for initial conditions in the regions highlighted in yellow, for which the algorithm converges to a limit cycle with K_1 corresponding to NE 3 and K_2 corresponding to NE 2 and *vice versa*, as shown in Figure 3.13.

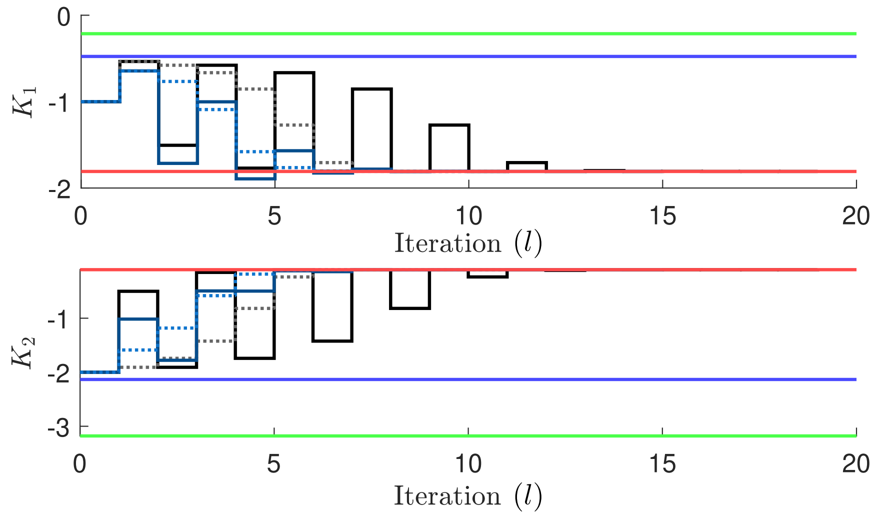


Figure 3.11: Update history of the *Lyapunov iterations* (3.72) (dark blue lines) and the *Riccati iterations* (3.75) (black lines). Both the synchronous (solid lines) and the asynchronous (dotted lines) versions are shown. The Nash equilibrium strategies in Table 3.4 are highlighted in blue (NE 1), green (NE 2) and red (NE 3). (© 2022 IEEE)

Table 3.4: Parameters corresponding to the three feedback Nash equilibria of the considered game rounded to four decimal places.

Equilibrium	$P_1^\star$	$P_2^\star$	$K_1^\star$	$K_2^\star$
NE 1	2.0066	29.1606	−0.4771	−2.1339
NE 2	0.9197	44.4309	−0.2138	−3.1793
NE 3	7.9681	1.4540	−1.8084	−0.1016

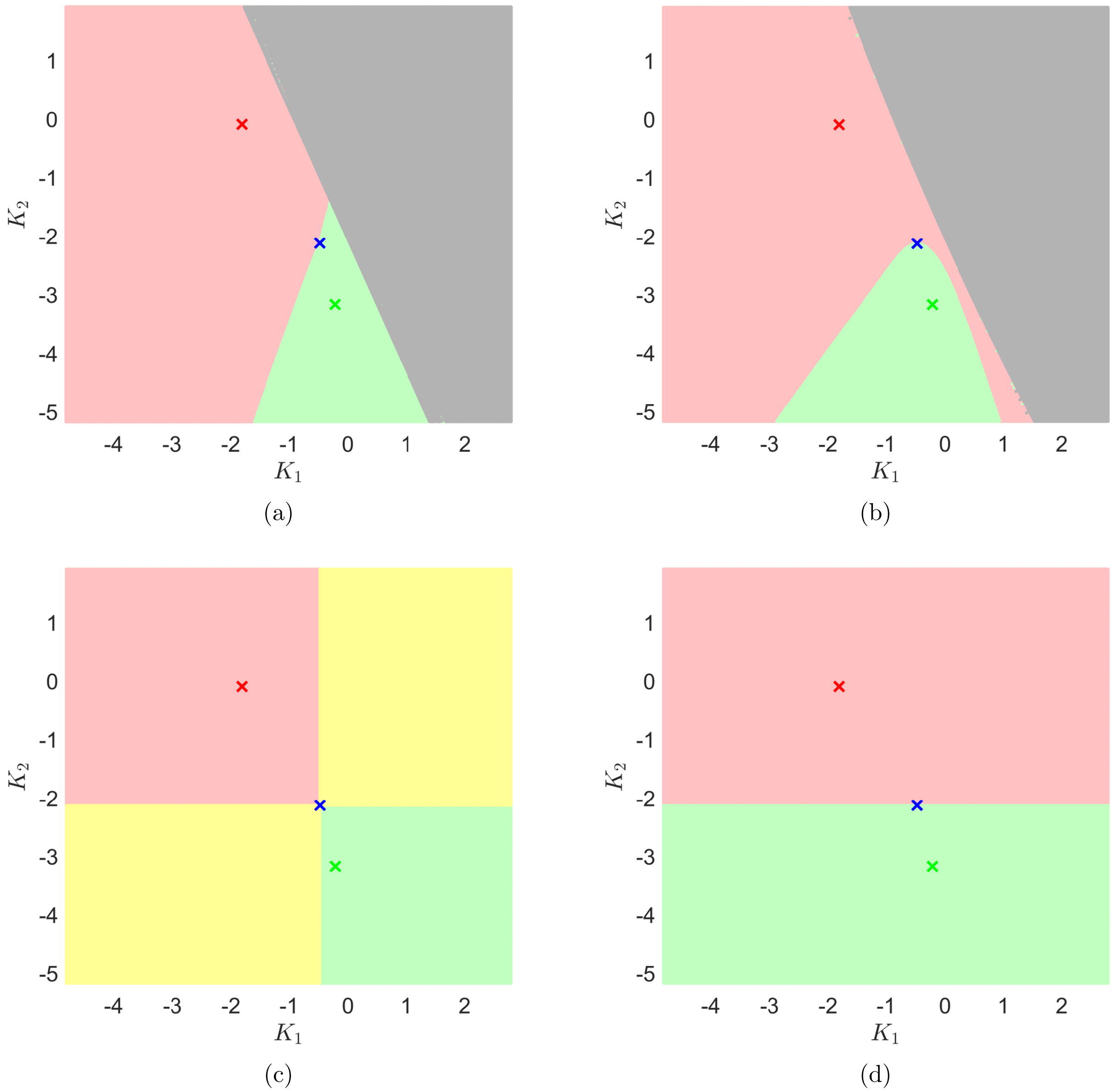


Figure 3.12: Regions of the K_1-K_2 space starting from which the synchronous Lyapunov iterations (3.72), $\sigma = s$ (a), asynchronous Lyapunov iterations (3.72), $\sigma = a$ (b), synchronous Riccati iterations (3.75), $\sigma = s$ (c) and asynchronous Riccati iterations (3.75), $\sigma = a$ (d) converge to NE 1 (blue), NE 2 (green) and NE 3 (red). The yellow regions highlight initial conditions for which the synchronous Riccati iterations (3.75), $\sigma = s$, converge to a limit cycle. The grey regions highlight initial conditions that do not converge to any of the three equilibria or a limit cycle.

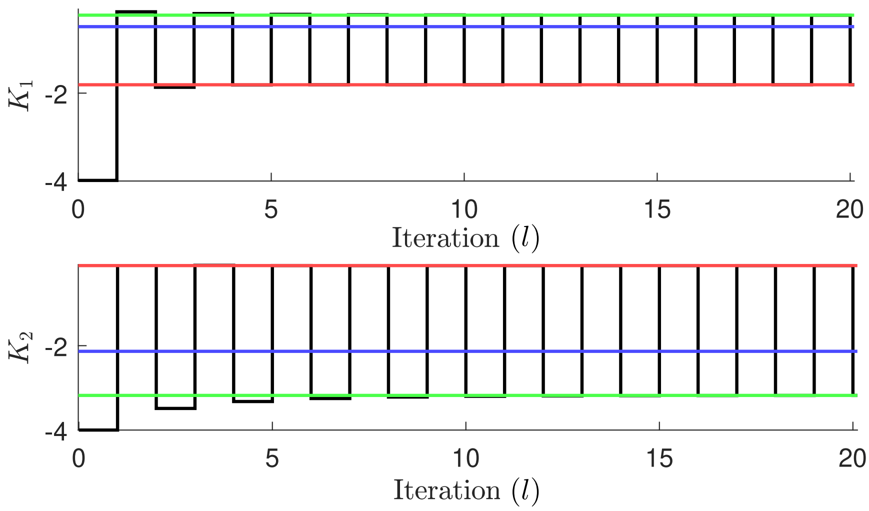


Figure 3.13: Update history of the synchronous Riccati iterations (3.75), $\sigma = s$, with initial conditions $K_1^{(0)} = K_2^{(0)} = -4$. The Nash equilibrium strategies in Table 3.4 are highlighted in blue (NE 1), green (NE 2) and red (NE 3).

Four-player example

Consider the numerical example used in [74, 78], namely, consider the game defined by the dynamics (3.1) with $N = 4$,

$$A = \begin{bmatrix} 0.995 & 0.09983 \\ -0.09983 & 0.995 \end{bmatrix}, \; B_1 = \begin{bmatrix} 0.2047 \\ 0.08984 \end{bmatrix}, \; B_2 = \begin{bmatrix} 0.2147 \\ 0.2895 \end{bmatrix}, \; B_3 = \begin{bmatrix} 0.2097 \\ 0.1897 \end{bmatrix}, \; B_4 = \begin{bmatrix} 0.2 \\ 0.1 \end{bmatrix}$$

and the cost functionals (3.2), for $i = 1, \dots, 4$, with

$$Q_{11} = Q_{22} = Q_{33} = Q_{44} = I_2,$$

$$R_{11} = R_{22} = R_{33} = R_{44} = R_{12} = R_{14} = R_{23} = R_{31} = 1,$$

and

$$R_{13} = R_{21} = R_{24} = R_{32} = R_{34} = R_{41} = R_{42} = R_{43} = 0.$$

Consider the initial guess

$$K_1^{(0)} = \begin{bmatrix} -2.2570 & 1.1761 \end{bmatrix}, \; K_2^{(0)} = \begin{bmatrix} 1.1629 & -2.5437 \end{bmatrix},$$

$$K_3^{(0)} = \begin{bmatrix} -0.5465 & -0.6844 \end{bmatrix}, \; K_4^{(0)} = \begin{bmatrix} -1.9842 & 0.9127 \end{bmatrix},$$

as in [74]. The four algorithms presented in Section 3.4.2 are compared to [73, Algorithm 1] (extended to the N-player case). Note that the synchronous Lyapunov iterations (3.72), $\sigma = s$, are equivalent to

[74, Algorithm 1]. All five algorithms converge to

$$K_1^\star = \begin{bmatrix} -0.6918 & 0.1601 \end{bmatrix}, \; K_2^\star = \begin{bmatrix} 0.0052 & -0.6095 \end{bmatrix},$$

$$K_3^\star = \begin{bmatrix} -0.3953 & -0.1560 \end{bmatrix}, \; K_4^\star = \begin{bmatrix} -0.4239 & 0.0442 \end{bmatrix},$$

which satisfy (3.4), (3.5), $i = 1, \dots, 4$, and (3.6) with

$$P_1^\star = \begin{bmatrix} 5.6345 & -2.6678 \\ -2.6678 & 4.2977 \end{bmatrix}, \; P_2^\star = \begin{bmatrix} 4.3227 & -2.1610 \\ -2.1610 & 4.2364 \end{bmatrix},$$

$$P_3^\star = \begin{bmatrix} 4.8907 & -1.9054 \\ -1.9054 & 3.2167 \end{bmatrix}, \; P_4^\star = \begin{bmatrix} 3.8844 & -1.7237 \\ -1.7237 & 3.0592 \end{bmatrix},$$

where all values are rounded to four decimal places. The number of iterations taken by the algorithms until the convergence criterion $\max_i \left(\left\| K_i^{(l+1)} - K_i^{(l)} \right\| \right) \leq 10^{-5}$ is reached is reported in Table 3.5.

Table 3.5: Number of iterations until convergence for the different Nash equilibrium-finding algorithms.

Algorithm	[73, Alg. 1]	(3.72), $\sigma = s$ ([74, Alg. 1])	(3.72), $\sigma = a$	(3.75), $\sigma = s$	(3.75), $\sigma = a$
No. of iterations	12	30	11	31	10

3.5 Conclusion

Motivated by the challenges associated with determining feedback Nash equilibria of infinite-horizon nonzero-sum LQ discrete-time dynamic games, the corresponding class of scalar games, i.e. games involving dynamics in which the state and the input of each player are scalar variables, is studied. A graphical representation of the conditions characterising feedback Nash equilibrium solutions is proposed. Via geometric arguments, this representation allows to characterise the possible number and properties of solutions in terms of the system and cost parameters. Additional insights are provided for the scalar two-player case. The results are illustrated via numerical examples.

Considering general LQ discrete-time dynamic games, a solution concept, which approximates exact Nash equilibria, is introduced - the more readily obtained $\epsilon_{\alpha,\beta}$-Nash equilibria, and the degree of approximation is discussed. The presented notion of approximate Nash equilibrium solution incorporates guarantees on the rate of convergence of the trajectories of the resulting closed-loop system and constitutes a discrete-time counterpart to the notions of approximate solution in [62] with some generalisations. The results are illustrated via a simulation example involving macroeconomic policy design.

Four iterative algorithms for finding feedback Nash equilibrium strategies of LQ discrete-time dynamic games are proposed. The algorithms are based on update laws involving the solution of uncoupled Lyapunov or Riccati equations. For each type a synchronous (i.e. all players update their strategy simultaneously) and an asynchronous (i.e. each player's update takes the previous players' updates at the same iteration step into account) version are presented. Local convergence conditions are provided. The performance of the algorithms is illustrated and compared to alternative algorithms in the literature via two numerical examples.

Chapter 4

Direct data-driven control methods

Direct data-driven control methods, which aim to control a system directly using data, without explicitly identifying a system model, have recently attracted significant interest, as discussed in Chapter 2. This chapter builds on the direct data-driven control framework originally introduced in [95]. In Section 4.1, it is extended to the class of LTV systems. Motivated by the fact that optimality criteria are often not known a priori in practical applications, a method to represent not only the system dynamics, but also quadratic cost functions in the context of optimal control directly using finite-length, non-optimal data of the input, state and a performance variable is derived in Section 4.2.

4.1 Data-driven control of linear time-varying systems

Time-varying systems arise in a variety of practical applications, for instance high-speed aircraft experiencing varying aerodynamic coefficients and electrical circuits or chemical plants subject to changing behaviour. LTV models also result when linearising nonlinear systems around a trajectory or time-varying operating point [172]. In this section, a direct data-driven control design method is introduced for discrete-time LTV systems with unknown dynamics. The demand for data-driven or model-free control approaches for LTV systems is apparent in the literature, see e.g. [191, 192, 193, 194]. Differently from these works, the control design approach presented herein does not require any iterative procedures or machine learning techniques. Extending the data-driven framework originally presented in [95] for LTI systems to the LTV setting, the closed-loop system under state-feedback and the controller are parameterised directly using an ensemble of input-state date. In contrast to the related results [127], [130], the presented data-driven methods are applicable to (linear) *arbitrarily* time-varying systems and do not rely on any assumptions or prior knowledge of the system structure or parameter variation. However, it is shown how such knowledge can be exploited for the special case of periodically time-varying systems. Both noise-free LTV systems and LTV systems affected by measurement as well as process noise are considered. The noisy data results presented herein are inspired by [95, 146], and can be considered as an LTV equivalent. The main difference apart from the extension to LTV systems - which itself introduces new challenges and requires a different approach to parametrise unknown systems - is that *both* measurement and process noise are incorporated in a single formulation and that the behaviour of the system in closed-loop with feedback on the noisy state measurements is studied.

The remainder of this section is organised as follows. Section 4.1.1 defines the considered problem. In Section 4.1.2 noise-free LTV systems are considered and methods to design state-feeback controllers guaranteeing a decreasing bound on the closed-loop trajectories or solving the time-varying LQR problem via data-dependent convex optimisation problems are introduced. Section 4.1.3 focuses on LTV systems affected by process and measurement noise and addresses the problem of designing controllers with robustness guarantees directly using noisy data. In Section 4.1.4 approaches are proposed to overcome challenges which may arise for control problems over large time horizons. In Section 4.1.5, the results are specialised to the class of linear periodically time-varying systems. The efficacy and relevance of the results is demonstrated via numerical and practically motivated examples in Section 4.1.6.

4.1.1 Problem formulation

Consider a discrete-time LTV system, described by

$$x(k+1) = A(k)x(k) + B(k)u(k) + d(k), \tag{4.1a}$$

where $x \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{R}^m$ is the control input, $d \in \mathbb{R}^n$ denotes an unknown additive system noise and $A(k)$ and $B(k)$ denote the *unknown* time-varying dynamics and input matrices of appropriate dimensions, respectively. Suppose that available state measurements $\zeta \in \mathbb{R}^n$ are corrupted by measurement noise $v \in \mathbb{R}^n$, i.e.

$$\zeta(k) = x(k) + v(k). \tag{4.1b}$$

Our objective is to design controllers of the form

$$u(k) = K(k)\zeta(k), \tag{4.2}$$

for the unknown LTV system (4.1) based solely on measurements of the (noisy) state and input of the system, such that certain guarantees hold for the resulting closed-loop system

$$x(k+1) = \Big(A(k) + B(k)K(k)\Big)x(k) + B(k)K(k)v(k) + d(k). \tag{4.3}$$

If $d(k) = 0$ and $\zeta(k) = x(k)$, for all k, i.e. in the *noise-free* case, the dynamics of the unknown LTV system (4.1) simplify to

$$x(k+1) = A(k)x(k) + B(k)u(k), \tag{4.4}$$

and the closed-loop system under state-feedback with

$$u(k) = K(k)x(k), \tag{4.5}$$

is described by

$$x(k+1) = \Big(A(k) + B(k)K(k)\Big)x(k). \tag{4.6}$$

This section concerns the direct data-driven solution of several classical control problems involving the unknown systems (4.4) or (4.1). Consider the following standing assumption.

Assumption 4.1.1. It is possible to gather an ensemble of $L \in \mathbb{N}$ input-state data sequences capturing the *same time-varying behaviour* of the unknown LTV system over $T+1$ time instances, with $T \in \mathbb{N}$, i.e. if data sequence j covers the time interval $k = k_j, \dots, k_j + T$, for $j = 1, \dots, L$, then for all $l = 1, \dots, L$,[1]

$$\{A(k_j), \dots, A(k_j + T - 1)\} = \{A(k_l), \dots, A(k_l + T - 1)\},$$

and

$$\{B(k_j), \dots, B(k_j + T - 1)\} = \{B(k_l), \dots, B(k_l + T - 1)\}.$$

Remark 4.1.1. Assumption 4.1.1 is similar to requirements commonly encountered in ensemble methods for LTV system identification (see e.g. [195]). It is readily satisfied by systems arising in a variety of applications, including biomedical systems, nonlinear systems linearised along a trajectory and periodically time-varying systems, which are addressed in Section 4.1.5. Variations in environmental conditions that may result in different time-variations affecting each experiment in an ensemble can be considered as process noise, which is addressed in Section 4.1.3.

The L input-state data sequences can be obtained via a sequence of physical experiments or via simulations[2]. Considering the system (4.4), let $u_{d,j,[0,T-1]}$, $x_{d,j,[0,T]}$, represent input-state data collected during the j^{th} experiment, for $j = 1, \dots, L$. While the specific experiment is indicated by the subscript j, the subscript d highlights that the input-state sequences contain *measured data* samples. Consider the matrices

[1]Throughout this section, each interval capturing the time-variation of interest is referred to as $k = 0, \dots, T$, i.e. $k_j = 0$, for $j = 1, \dots, L$.

[2]Herein, the act of data collection is referred to as "experiment", regardless of whether the data is collected via physical experiments or simulations.

$$X(k) = \begin{bmatrix} x_{d,1}(k) & x_{d,2}(k) & \dots & x_{d,L}(k) \end{bmatrix}, \tag{4.7a}$$

for $k = 0, \dots, T$, and

$$U(k) = \begin{bmatrix} u_{d,1}(k) & u_{d,2}(k) & \dots & u_{d,L}(k) \end{bmatrix}, \tag{4.7b}$$

for $k = 0, \dots, T-1$, which combine the data from all L experiments at each time step. Note that the data matrices $X(k)$ and $U(k)$ satisfy

$$X(k+1) = A(k)X(k) + B(k)U(k) = [A(k)\ B(k)] \begin{bmatrix} X(k) \\ U(k) \end{bmatrix}, \tag{4.8}$$

for $k = 0, \dots, T-1$. Similarly, considering the system (4.1), let $u_{d,j,[0,T-1]}$, $\zeta_{d,j,[0,T]}$, denote input-output data collected during the j^{th} experiment, for $j = 1, \dots, L$. The data is arranged to form the matrices

$$Z(k) = \begin{bmatrix} \zeta_{d,1}(k) & \zeta_{d,2}(k) & \dots & \zeta_{d,L}(k) \end{bmatrix}, \tag{4.9}$$

for $k = 0, \dots, T$, representing the ensemble of noisy state measurements, and (4.7b), for $k = 0, \dots, T-1$, representing the ensemble of input measurements. Consider also a similar stacking of the corresponding samples of the state x_d, the measurement noise v_d and the system noise d_d (all of which are *not measured*) associated with the ensemble of experiments, represented by the matrices (4.7a) and

$$V(k) = \begin{bmatrix} v_{d,1}(k) & v_{d,2}(k) & \dots & v_{d,L}(k) \end{bmatrix}, \tag{4.10}$$

for $k = 0, \dots, T$, and

$$D(k) = \begin{bmatrix} d_{d,1}(k) & d_{d,2}(k) & \dots & d_{d,L}(k) \end{bmatrix}, \tag{4.11}$$

for $k = 0, \dots, T-1$, respectively. Note that

$$\begin{aligned} X(k+1) &= A(k)X(k) + B(k)U(k) + D(k), \\ Z(k) &= X(k) + V(k). \end{aligned} \tag{4.12}$$

In the following sections, the matrices $X(k)$, $U(k)$ or $Z(k)$, $U(k)$ are utilised to design controllers with trajectory boundedness, performance and robustness guarantees for the unknown systems (4.4) and (4.1), respectively, using data only.

4.1.2 The noise-free case

This section focuses on the noise-free unknown LTV system (4.4). Employing measured data, feedback controllers of the form (4.5) are designed via the solution of convex optimisation problems involving LMI constraints.

Data-driven system representation

Consider the closed-loop system under state-feedback (4.6). In the following result a representation of the system and the control gain, which directly uses the data matrices defined in Section 4.1.1, is proposed.

Corollary 4.1.1. *Suppose the rank condition*[3]

$$\operatorname{rank}\left(\begin{bmatrix} X(k) \\ U(k) \end{bmatrix}\right) = n + m, \tag{4.13}$$

holds for $k = 0, \ldots, T-1$. *Then, the closed-loop system* (4.6) *can equivalently be represented as*

$$x(k+1) = X(k+1)G(k)x(k), \tag{4.14}$$

where $G(k) \in \mathbb{R}^{L \times n}$ *satisfies*

$$\begin{bmatrix} I_n \\ K(k) \end{bmatrix} = \begin{bmatrix} X(k) \\ U(k) \end{bmatrix} G(k), \tag{4.15}$$

for $k = 0, \ldots, T-1$.

Proof. Analogous to the proof of [95, Theorem 2], but considering the alternative data matrices combining data from an ensemble of experiments for the LTV system (4.4), note that if (4.13) holds, for $k = 0, \ldots, T-1$, then there exists $G(k)$ satisfying (4.15), for $k = 0, \ldots, T-1$. Hence,

$$A(k) + B(k)K(k) = \begin{bmatrix} A(k) & B(k) \end{bmatrix} \begin{bmatrix} I_n \\ K(k) \end{bmatrix} = \begin{bmatrix} A(k) & B(k) \end{bmatrix} \begin{bmatrix} X(k) \\ U(k) \end{bmatrix} G(k) = X(k+1)G(k), \tag{4.16}$$

which gives (4.14). □

In Corollary 4.1.1 the sequence of control gains $K(k)$ is parameterised using data through the identity (4.15). Hence, the matrices $G(k)$, for $k = 0, \ldots, T-1$, can be seen as decision variables, which can be used for identification-free design of state-feedback controllers.

Remark 4.1.2. To utilise the data-driven system representation (4.14), (4.15), it is required that (4.13) holds for all $k = 0, \ldots, T-1$. Thus, each input-state data sequence j, $j = 1, \ldots, L$, in the ensemble must start at different initial conditions $x_{d,j}(0)$. If this is infeasible, a common starting point can be considered as the state at $k = -1$ and different inputs can be applied for each experiment to obtain different state data at $k = 0$.

[3] The condition (4.13) can always be verified from the measured data. A necessary condition for (4.13) to hold is that $L \geq n + m$.

Bounded closed-loop trajectories

Consider the problem of controlling the LTV system (4.4) over a finite time horizon, with the aim of ensuring that the closed-loop trajectories remain close to the equilibrium throughout the considered horizon. A solution to this problem is provided in the following result.

Theorem 4.1.1. *Consider the system* (4.4) *and suppose an ensemble of input-state data is available to form the matrices* (4.7a), (4.7b), *such that the rank condition* (4.13) *holds, for* $k = 0, \ldots, T-1$. *Any sequences of matrices* $Y(k)$, $P(k) = P(k)^\top$ *satisfying*

$$\begin{bmatrix} P(k+1) - I_n & X(k+1)Y(k) \\ Y(k)^\top X(k+1)^\top & P(k) \end{bmatrix} \succeq 0, \tag{4.17a}$$

$$X(k)Y(k) = P(k), \tag{4.17b}$$

for $k = 0, \ldots, T-1$, *and*

$$\eta I_n \preceq P(k) \preceq \rho I_n, \tag{4.17c}$$

for $k = 0, \ldots, T$, *where* $\eta \geq 1$ *and* $\rho > \eta$ *are finite constants, are such that the trajectories of the system* (4.6), *with*

$$K(k) = U(k)Y(k)P(k)^{-1}, \tag{4.18}$$

for $k = 0, \ldots, T-1$, *satisfy*

$$\|x(k)\| \leq \sqrt{\frac{\rho}{\eta}} \left(1 - \frac{1}{\rho}\right)^{\frac{k}{2}} \|x(0)\|, \tag{4.19}$$

for $k = 0, \ldots, T$.

Proof. To demonstrate the claim consider the adjoint equation (see, e.g. [196, Section 3.1]) of the closed-loop system (4.6). Namely, consider

$$\xi(j) = A_{cl}(j)^\top \xi(j+1), \tag{4.20}$$

and note that the solution to (4.20) starting from $\xi(k)$ is

$$\xi(j) = S_t(k,j)^\top \xi(k), \tag{4.21}$$

for $j \leq k$, where

$$S_t(k,j) = \begin{cases} A_{cl}(k-1)A_{cl}(k-2)\ldots A_{cl}(j), & \text{for } j < k, \\ I_n, & \text{for } j = k, \end{cases}$$

denotes the state transition matrix corresponding to the closed-loop system (4.6), and where $A_{cl}(k) =$

$A(k) + B(k)K(k)$. Let $\xi(k) \neq 0$ and suppose it is possible to determine a sequence of matrices $P(k)$ satisfying the condition (4.17c) and

$$A_{cl}(k)P(k)A_{cl}(k)^\top - P(k+1) + I_n \preceq 0, \tag{4.22}$$

for $k = 0, \dots, T-1$. Consider the quadratic function $\bar{V}_j := \bar{V}(j, \xi(j)) := \xi(j)^\top P(j)\xi(j)$, for $j = 0, \dots, T$. It follows from (4.20), (4.22) and (4.17c) that

$$\bar{V}_{j+1} - \bar{V}_j \geq \|\xi(j+1)\|^2 \geq \frac{1}{\rho}\xi(j+1)^\top P(j+1)\xi(j+1),$$

for $j = 0, \dots, T-1$, and hence

$$\xi(j)^\top P(j)\xi(j) \leq \left(1 - \frac{1}{\rho}\right)^{k-j} \xi(k)^\top P(k)\xi(k),$$

for $j = 0, \dots, T$, $j \leq k \leq T$. It then follows from (4.17c) that

$$\eta\|\xi(j)\|^2 \leq \rho\left(1 - \frac{1}{\rho}\right)^{k-j} \|\xi(k)\|^2,$$

which, using (4.21), in turn yields

$$\|S_t(k,j)^\top \xi(k)\|^2 \leq \frac{\rho}{\eta}\left(1 - \frac{1}{\rho}\right)^{k-j} \|\xi(k)\|^2,$$

and

$$\|S_t(k,j)^\top\|^2 = \|S_t(k,j)\|^2 \leq \frac{\rho}{\eta}\left(1 - \frac{1}{\rho}\right)^{k-j}, \tag{4.23}$$

for $j = 0, \dots, T$, $j \leq k \leq T$. Noting that $x(k) = S_t(k,j)x(j)$, for $k \geq j$, (4.23) implies

$$\begin{aligned} \|x(k)\|^2 = \|S_t(k,j)x(j)\|^2 &\leq \|S_t(k,j)\|^2\|x(j)\|^2 \\ &\leq \frac{\rho}{\eta}\left(1 - \frac{1}{\rho}\right)^{k-j} \|x(j)\|^2, \end{aligned} \tag{4.24}$$

for $j = 0, \dots, T$, $j \leq k \leq T$. Letting $j = 0$, this yields (4.19). Finally, using (4.14), (4.15), defining $Y(k) := G(k)P(k)$, and via the Schur complement, (4.17a) is equivalent to (4.22), if (4.17b) holds and the control gain is chosen as in (4.18). □

Optimal control

Consider the system (4.4) and the problem of finding the optimal control sequence $\{u^\star(0), u^\star(1), \dots, u^\star(T-1)\}$ as a function of the state, which minimises the quadratic cost functional

$$J\left(x(0), u(\cdot)\right) = x(T)^\top Q_f x(T) + \sum_{k=0}^{T-1} \Big(x(k)^\top Q(k) x(k) + u(k)^\top R(k) u(k) \Big), \tag{4.25}$$

over the time horizon $T \in \mathbb{N}$, starting from the initial condition $x(0) = x_0$, with $Q_f = Q_f^\top \succeq 0$, $Q(k) = Q(k)^\top \succeq 0$ and $R(k) = R(k)^\top \succ 0$, for $k = 0, \dots, T-1$. The solution to this finite-horizon LQR problem is well-known to be given by

$$u^\star(k) = K^\star(k) x(k), \tag{4.26}$$

with the time-varying gain matrix $K^\star(k)$ given by

$$K^\star(k) = -\Big(R(k) + B(k)^\top \bar{P}(k+1) B(k) \Big)^{-1} B(k)^\top \bar{P}(k+1) A(k), \tag{4.27}$$

where the symmetric and positive-definite matrix $\bar{P}(k)$, with $\bar{P}(T) = Q_f$, is the solution of the difference Riccati equation

$$\begin{aligned} \bar{P}(k) &= Q(k) + A(k)^\top \bar{P}(k+1) A(k) \\ &\quad - A(k)^\top \bar{P}(k+1) B(k) \Big(R(k) + B(k)^\top \bar{P}(k+1) B(k) \Big)^{-1} B(k)^\top \bar{P}(k+1) A(k), \end{aligned} \tag{4.28}$$

for $k = 0, \dots, T-1$ (see e.g. [197, Section 4.1]).

To derive a solution to the optimal control problem in terms of data only, an equivalent formulation of the LQR problem is considered - the covariance selection problem (see [198]). Building upon this formulation it can be shown (see [199, Sections 2.1, 3.1][4]) that solving the finite-horizon LQR problem is equivalent to solving the optimisation problem

$$\min_{\mathcal{S}, \mathcal{K}, \mathcal{O}} \quad \mathbf{Tr}\left(Q_f S(T)\right) + \sum_{k=0}^{T-1} \Big(\mathbf{Tr}\left(Q(k) S(k)\right) + \mathbf{Tr}\left(O(k)\right) \Big) \tag{4.29a}$$

$$\text{s.t.} \quad S(0) \succeq I_n, \tag{4.29b}$$

$$S(k+1) - I_n - \left(A(k) + B(k)K(k)\right) S(k) \left(A(k) + B(k)K(k)\right)^\top \succeq 0, \tag{4.29c}$$

$$O(k) - R(k)^{1/2} K(k) S(k) K(k)^\top R(k)^{1/2} \succeq 0, \tag{4.29d}$$

[4]While [199] concerns LTI systems, all arguments readily extend to the time-varying case.

for $k = 0, \dots, T-1$, with

$$\mathcal{S} := \{S(1), \dots, S(T)\}, \ \mathcal{O} := \{O(0), \dots, O(T-1)\} \ \text{and} \ \mathcal{K} := \{K(0), \dots, K(T-1)\}.$$

The optimal gain sequence for the feedback law (4.26) is given by the solution $\mathcal{K}^\star$ to (4.29).

In the following result, this formulation of the LQR problem is combined with the data-driven system representation introduced in Corollary 4.1.1 to formulate the *time-varying* LQR problem as a data-dependent SDP.

Theorem 4.1.2. *Consider the system* (4.4) *and suppose an ensemble of input-state data is available to form the matrices* (4.7a), (4.7b), *such that the rank condition* (4.13) *holds, for* $k = 0, \dots, T-1$. *The optimal state-feedback control gain sequence* $\{K^\star(0), K^\star(1), \dots, K^\star(T-1)\}$ *solving the finite-horizon LQR problem with* $u^\star(k) = K^\star(k)x(k)$ *is given by*

$$K^\star(k) = U(k)H^\star(k)S^\star(k)^{-1}, \tag{4.30}$$

for $k = 0, \dots, T-1$, *with* $H^\star(k)$ *and* $S^\star(k)$ *the solution of*

$$\min_{\mathcal{S}, \mathcal{H}, \mathcal{O}} \quad \mathbf{Tr}\left(Q_f S(T)\right) + \sum_{k=0}^{T-1} \Big(\mathbf{Tr}\left(Q(k)S(k)\right) + \mathbf{Tr}\left(O(k)\right) \Big) \tag{4.31a}$$

$$\text{s.t.} \quad S(0) \succeq I_n, \tag{4.31b}$$

$$\begin{bmatrix} S(k+1) - I_n & X(k+1)H(k) \\ H(k)^\top X(k+1)^\top & S(k) \end{bmatrix} \succeq 0, \tag{4.31c}$$

$$\begin{bmatrix} O(k) & R(k)^{1/2}U(k)H(k) \\ H(k)^\top U(k)^\top R(k)^{1/2} & S(k) \end{bmatrix} \succeq 0, \tag{4.31d}$$

$$S(k) = X(k)H(k), \tag{4.31e}$$

for $k = 0, \dots, T-1$, *where*

$$\mathcal{S} = \{S(1), \dots, S(T)\}, \ \mathcal{H} = \{H(0), \dots, H(T-1)\} \ \textit{and} \ \mathcal{O} = \{O(0), \dots, O(T-1)\}.$$

Proof. The proof lies in demonstrating that the data-based problem (4.31) is equivalent to the model-based problem (4.29). This follows by introducing (4.14), (4.15) to the constraints, letting $H(k) := G(k)S(k)$, and taking the Schur complement of the nonlinear inequality constraints. □

4.1.3 Robustness to noise

In practice, both the measurements and/or the system dynamics may be subject to noise. This section focuses on designing feedback controllers for the general unknown LTV system (4.1) via the solution of convex optimisation problems involving LMI constraints.

Data-driven system representation

Towards designing controllers directly using noise corrupted data, an alternative data-driven system representation of the form (4.14), (4.15) is derived for the closed-loop system (4.3) in the following result.

Corollary 4.1.2. *Suppose the rank condition*

$$\mathit{rank}\left(\begin{bmatrix} Z(k) \\ U(k) \end{bmatrix}\right) = n + m, \tag{4.32}$$

holds for $k = 0, \ldots, T-1$*. Then, the dynamics matrix of the closed-loop system* (4.3) *can equivalently be represented as*

$$A(k) + B(k)K(k) = \Big(Z(k+1) - W(k)\Big)G(k), \tag{4.33}$$

where $G(k)$ *satisfies*

$$\begin{bmatrix} I_n \\ K(k) \end{bmatrix} = \begin{bmatrix} Z(k) \\ U(k) \end{bmatrix} G(k), \tag{4.34}$$

for $k = 0, \ldots, T-1$*, with*

$$W(k) = D(k) + V(k+1) - A(k)V(k). \tag{4.35}$$

Proof. The result follows via analogous steps to the proof of Corollary 4.1.1 and by recalling that the ensemble data matrices satisfy (4.12). □

As in Corollary 4.1.1, the matrices $G(k)$, for $k = 0, \ldots, T-1$, become the decision variables for control design. Unlike the noise-free case considered in Section 4.1.2, the data-driven system representation (4.33), (4.34) in the general case depends on the unknown matrix $W(k)$ as defined in (4.35), which contains both system[5] and noise information. However, if $W(k)$ lies in a specified uncertainty set for $k = 0, \ldots, T-1$, controllers with trajectory boundedness and performance guarantees can be designed via data-dependent convex programmes, as detailed below.

[5] As in the LTI case the appearance of $A(k)$ in (4.35) can be interpreted as a measure of the direction of the measurement noise, which contributes to the loss of information caused [95].

Bounded closed-loop trajectories

Consider the problem of controlling the LTV system (4.1) over a finite time horizon, with the aim of ensuring that the trajectories of the resulting closed-loop system (4.3) stay within a decreasing bound similar to (4.19). To this end, an alternative bound is derived, which is related to the notion of input-to-state stability (see e.g. [200]).

Lemma 4.1.1. *Suppose there exists $P(k) = P(k)^\top$ satisfying* (4.17c)*, for $k = 0, \dots, T$, and* (4.22) *for some $K(k)$, for $k = 0, \dots, T-1$. The state trajectories of the system* (4.3) *satisfy*

$$\|x(k)\| \leq \sqrt{\frac{\rho}{\eta}}\left(1-\frac{1}{\rho}\right)^{\frac{k}{2}} \|x(0)\| + \gamma_1\left(|v|_{k-1}, k\right) + \gamma_2\left(|d|_{k-1}, k\right), \tag{4.36}$$

for $k = 0, \dots, T$, with $\gamma_1(\cdot, k)$, $\gamma_2(\cdot, k)$ class $\mathcal{K}$-functions.

Proof. The state response of system (4.3) at time k is given by

$$x(k) = S_t(k,0)x(0) + \sum_{j=0}^{k-1} S_t(k-1,j)(B(j)K(j)v(j) + d(j)),$$

where $S_t(k,0)$ is the state transition matrix corresponding to (4.6) as defined in Section 4.1.2. From Theorem 4.1.1 it holds that if there exist $P(k) = P(k)^\top$, $K(k)$ satisfying (4.17c), for $k = 0, \dots, T$, and (4.22), for $k = 0, \dots, T-1$, then $\|S_t(k,0)\| \leq \sqrt{\frac{\rho}{\eta}}\left(1-\frac{1}{\rho}\right)^{\frac{k}{2}}$, for $k = 0, \dots, T$. Combined with properties of the operator norm this gives (4.36) with

$$\begin{aligned}\gamma_1(|v|_{k-1}, k) &= b\left(\sum_{j=0}^{k-1} \sqrt{\frac{\rho}{\eta}}\left(1-\frac{1}{\rho}\right)^{\frac{k-1-j}{2}} \|K(j)\|\right)|v|_{k-1},\\ \gamma_2(|d|_{k-1}, k) &= \left(\sum_{j=0}^{k-1} \sqrt{\frac{\rho}{\eta}}\left(1-\frac{1}{\rho}\right)^{\frac{k-1-j}{2}}\right)|d|_{k-1},\end{aligned} \tag{4.37}$$

where b denotes the upper bound on the singular values of $B(\cdot)$, i.e. $\|B(j)\| \leq b$ for $0 \leq j \leq k-1$. □

With the aim of designing controllers such that (4.36) holds, for $k = 0, \dots, T$, directly using noisy data, the following result combines the results of Lemma 4.1.1 and Corollary 4.1.2.

Theorem 4.1.3. *Consider the system* (4.1) *and suppose an ensemble of input-output data is available to form the matrices* (4.9)*,* (4.7b)*, such that the rank condition* (4.32) *holds, for $k = 0, \dots, T-1$. Suppose $W(k)$ satisfies*

$$\begin{bmatrix} I_n \\ W(k)^\top \end{bmatrix}^\top \begin{bmatrix} Q_r(k) & S_r(k) \\ S_r(k)^\top & R_r(k) \end{bmatrix} \begin{bmatrix} I_n \\ W(k)^\top \end{bmatrix} \succeq 0, \tag{4.38}$$

where $Q_r(k) \in \mathbb{R}^{n\times n}$, $S_r(k) \in \mathbb{R}^{n\times L}$ and $R_r(k) \prec 0 \in \mathbb{R}^{L\times L}$, for $k = 0, \dots, T-1$. Any sequences of *matrices* $Y(k)$*,* $P(k) = P(k)^\top$ *satisfying*

$$\begin{bmatrix} P(k+1) - I_n - Q_r(k) & S_r(k) & Z(k+1)Y(k) \\ S_r(k)^\top & -R_r(k) & Y(k) \\ Y(k)^\top Z(k+1)^\top & Y(k)^\top & P(k) \end{bmatrix} \succ 0, \tag{4.39a}$$

$$Z(k)Y(k) = P(k), \tag{4.39b}$$

for $k = 0, \dots, T-1$*, and* (4.17c)*, for* $k = 0, \dots, T$*, where* $\eta \geq 1$ *and* $\rho > \eta$ *are finite constants, are such that the trajectories of the system* (4.3)*, with* $K(k)$ *given by* (4.18)*, for* $k = 0, \dots, T-1$*, satisfy* (4.36)*, for* $k = 0, \dots, T$*.*

Proof. By Lemma 4.1.1, (4.36) holds for the trajectories of (4.3) if there exist $P(k) = P(k)^\top$ satisfying (4.17c), for $k = 0, \dots, T$, and (4.22), for $k = 0, \dots, T-1$. Using (4.33)-(4.35), and letting $Y(k) := G(k)P(k)$, (4.22) is equivalent to

$$\Big(Z(k+1) - W(k)\Big)Y(k)P(k)^{-1}Y(k)\Big(Z(k+1) - W(k)\Big)^\top - P(k+1) + I_n \preceq 0. \tag{4.40}$$

Via the concrete version of the full block S-procedure (see [201], [202]), (4.40) is satisfied if (4.38) holds and $P(k)$, $Y(k)$ satisfy

$$\begin{bmatrix} \star & \star \\ \star & \star \\ \hline \star & \star \\ \star & \star \end{bmatrix}^\top \left[\begin{array}{cc|cc} -P(k) & 0 & 0 & 0 \\ 0 & P(k+1) - I_n & 0 & 0 \\ \hline 0 & 0 & -Q_r(k) & S_r(k) \\ 0 & 0 & S_r(k)^\top & -R_r(k) \end{array}\right] \begin{bmatrix} (Z(k+1)G(k))^\top & G(k)^\top \\ I_n & 0 \\ \hline I_n & 0 \\ 0 & I_L \end{bmatrix} \succ 0. \tag{4.41}$$

This quadratic form can be transformed into the LMI (4.39a) by performing the matrix multiplication, applying the Schur complement and a congruence transformation with $\operatorname{diag}(I_{n+L}, P(k))$. The constraint (4.39b) stems from the upper row block of (4.34). The lower row block of (4.34) is satisfied if the control gain $K(k)$ is chosen as in (4.18), for $k = 0, \dots, T-1$. □

Theorem 4.1.3 is the robust equivalent of Theorem 4.1.1. If the measured data is such that $W(k)$ lies in the uncertainty set (4.38), for $k = 0, \dots, T-1$, then controllers designed using the result of Theorem 4.1.3 are such that the trajectories of the closed-loop system (4.3) satisfy (4.36). Note that the uncertainty bound (4.38) is required to hold only for the measured data used for the system representation (4.33)-(4.35). Subsequently, (4.36) is satisfied by the trajectories of (4.3) for *arbitrary*, bounded noise inputs $d(k)$, $k = 0, \dots, T-1$, and $v(k)$, $k = 0, \dots, T$. Quantifying the trajectory bound (4.36) requires knowledge of b and the upper bound on the norm of the noise vectors, $|v|_{T-1}$ and $|d|_{T-1}$. Similarly, to verify (4.38), $k = 0, \dots, T-1$, knowledge of an upper bound on the unknown matrix

$A(k)$, for $k = 0, \ldots, T-1$, and the matrices $V(k)$, for $k = 0, \ldots, T$, and $D(k)$, for $k = 0, \ldots, T-1$, i.e. the ensembles of (unmeasured) samples of measurement and process noise corresponding to the collected input-output data, is required,[6] due to the definition of $W(k)$ in (4.35).

Remark 4.1.3. In the absence of measurement noise, i.e. if $v(k) = 0$, for $k = 0, \ldots, T$, (4.38) becomes a bound on $D(k)$ (the ensemble of process noise samples corresponding to the measured input-output data), which is similar to the bound on the noise data introduced in [146] for LTI systems subject to process noise only.[7] In this measurement noise free case, the dependence on the unknown matrix $A(k)$ disappears and only an upper bound on the process noise affecting the measured data is required to verify the uncertainty condition (4.38).

Remark 4.1.4. The matrices $Q_r(k)$, $S_r(k)$ and $R_r(k)$ in (4.38) are chosen by the user. This makes the quadratic bound (4.38) a flexible condition, which contains many practical bounds as special cases. Examples include a bound on the maximum singular value of $W(k)$, for $k = 0, \ldots, T-1$, as well as individual sample bounds on $w_d(k) = A(k)v_d(k) - v_d(k+1) - d_d(k)$, or bounds on the sample covariance across an ensemble of data sequences at each time step (see [146, 147] for details on the LTI equivalents). The choice $Q_r(k) = Z(k+1)Z(k+1)^\top$, $S_r(k) = 0$ and $R_r(k) = -\gamma(k)I_L$, for some $\gamma(k) > 0 \in \mathbb{R}$, gives the signal-to-noise ratio condition

$$W(k)W(k)^\top \preceq \frac{1}{\gamma(k)} Z(k+1)Z(k+1)^\top, \tag{4.42}$$

for $k = 0, \ldots, T-1$. This condition is similar (apart from being required to hold at each time step) to the condition presented in [95, Assumption 2] for LTI systems and represents a measure of the loss of information caused by the noise.

Robust performance

Consider the problem of designing controllers of the form (4.2) for the (unknown) LTV system (4.1), such that the closed-loop system (4.3) fulfils a disturbance attenuation condition. To this end, consider the performance output

$$\begin{aligned} z(k) &= C(k)x(k) + D_u(k)u(k) + D_d(k)d(k), \\ z_f &= C(T)x(T), \end{aligned} \tag{4.43}$$

[6]While the system dynamics and noise are assumed unknown, for many practical applications it is expected that reasonable upper bounds on these quantities can be estimated [182, Chapter 8].

[7]Note that in the LTV case (4.38) is required to hold at each time step, for $k = 0, \ldots, T-1$.

for $k = 0, \dots, T-1$, where $z \in \mathbb{R}^q$, $z_f \in \mathbb{R}^q$ and $C(T),\ C(k) \in \mathbb{R}^{q\times n}$, $D_u(k) \in \mathbb{R}^{q\times m}$, $D_d(k) \in \mathbb{R}^{q\times n}$ are known matrices. This results in the closed-loop system

$$x(k+1) = A_{cl}(k)x(k) + E_{cl}(k)\bar{w}(k), \tag{4.44a}$$

$$z(k) = C_{cl}(k)x(k) + D_{cl}(k)\bar{w}(k), \tag{4.44b}$$

$$z_f = C(T)x(T), \tag{4.44c}$$

$$\zeta(k) = x(k) + v(k), \tag{4.44d}$$

for $k = 0, \dots, T-1$, where

$$A_{cl}(k) := A(k) + B(k)K(k), \qquad E_{cl}(k) := \begin{bmatrix} B(k) & I_n \end{bmatrix},$$
$$C_{cl}(k) := C(k) + D_u(k)K(k), \qquad D_{cl}(k) := \begin{bmatrix} D_u(k) & D_d(k) \end{bmatrix},$$
$$\bar{w}(k) := \begin{bmatrix} \bar{v}(k)^\top & d(k)^\top \end{bmatrix}^\top, \qquad \bar{v}(k) := K(k)v(k).$$

Regarding $\bar{w}(k) \in \mathbb{R}^{(m+n)}$ as the disturbance, consider the quadratic robust performance criterion

$$z_f^\top z_f + \sum_{k=0}^{T-1} \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix}^\top \begin{bmatrix} Q_p(k) & S_p(k) \\ S_p(k)^\top & R_p(k) \end{bmatrix} \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix} + \varepsilon \sum_{k=0}^{T-1} \bar{w}(k)^\top \bar{w}(k) \leq 0, \tag{4.45}$$

for all $\bar{w} \in \ell_2$, where $\varepsilon > 0$ and $Q_p(k) \in \mathbb{R}^{(m+n)\times(m+n)}$, $S_p(k) \in \mathbb{R}^{(m+n)\times q}$ and $R_p(k) \succeq 0 \in \mathbb{R}^{q\times q}$, for $k = 0, \dots, T-1$.

Remark 4.1.5. The quadratic performance criterion (4.45) is the finite-horizon equivalent to the performance criterion introduced in [201], [202] and it captures many popular robust performance measures. For example, the choice $Q_p(k) = -\bar{\gamma}^2 I_{(m+n)}$, $S_p(k) = 0$ and $R_p(k) = I_q$, with $\bar{\gamma} > 0 \in \mathbb{R}$, for $k = 0, \dots, T-1$, recovers the finite-horizon H_∞-control problem for discrete LTV systems (see e.g. [196]).

Assuming the performance index is invertible, let

$$\begin{bmatrix} \tilde{Q}_p(k) & \tilde{S}_p(k) \\ \tilde{S}_p(k)^\top & \tilde{R}_p(k) \end{bmatrix} = \begin{bmatrix} Q_p(k) & S_p(k) \\ S_p(k)^\top & R_p(k) \end{bmatrix}^{-1}, \tag{4.46}$$

and further assume $\tilde{Q}_p(k) \prec 0$. The following result provides a strategy to design controllers ensuring the trajectories of (4.44) satisfy (4.45). For further results regarding robust performance of LTV systems see e.g. [196, 203].

Lemma 4.1.2. *Suppose there exists a matrix sequence $\mathcal{P}(k) = \mathcal{P}(k)^\top \succ 0$ satisfying*

$$\begin{bmatrix} \star & \star \\ \star & \star \\ \hline \star & \star \\ \star & \star \end{bmatrix}^\top \left[\begin{array}{cc|cc} -\mathcal{P}(k) & 0 & 0 & 0 \\ 0 & \mathcal{P}(k+1) & 0 & 0 \\ \hline 0 & 0 & \tilde{Q}_p(k) & -\tilde{S}_p(k) \\ 0 & 0 & -\tilde{S}_p(k)^\top & \tilde{R}_p(k) \end{array}\right] \left[\begin{array}{cc} A_{cl}(k)^\top & C_{cl}(k)^\top \\ I_n & 0 \\ \hline E_{cl}(k)^\top & D_{cl}(k)^\top \\ 0 & I_q \end{array}\right] \succ 0, \tag{4.47a}$$

for $k = 0, \dots, T-1$, and

$$I_q - C(T)\mathcal{P}(T)C(T)^\top \succeq 0. \tag{4.47b}$$

The output $z(k)$ of the closed-loop system (4.44) *subject to the disturbance input $\bar{w}(k)$ and with initial condition $x(0) = 0$ satisfies the quadratic robust performance criterion* (4.45)*.*

Proof. The result follows from dissipativity arguments (see e.g. [204] for a definition of dissipativiy for discrete-time systems). Suppose there exists a non-negative storage function $\mathcal{V}\left(x(k)\right) : \mathbb{R}^n \to \mathbb{R}$, such that

$$\mathcal{V}\left(x(k+1)\right) - \mathcal{V}\left(x(k)\right) \leq -\varepsilon \bar{w}(k)^\top \bar{w}(k) - \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix}^\top \begin{bmatrix} Q_p(k) & S_p(k) \\ S_p(k)^\top & R_p(k) \end{bmatrix} \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix}, \tag{4.48}$$

for $k = 0, \dots, T-1$, with $\mathcal{V}(0) = 0$, and

$$\mathcal{V}(T) \geq z_f^\top z_f. \tag{4.49}$$

Then, summing (4.48) from $k = 0$ to $k = T-1$ and using (4.49) gives (4.45). With the choice $\mathcal{V}\left(x(k)\right) = x(k)^\top \mathcal{P}(k)^{-1} x(k)$ and by using the dualisation lemma [201, Lemma 4.9], (4.48) and (4.49) are satisfied if the conditions (4.47a) and (4.47b) hold. □

With the aim of designing controllers, such that (4.45) holds directly using noisy data, consider (4.33)-(4.35) introduced in Corollary 4.1.2. A complication arises due to the fact that both measurement noise and process noise are considered. Namely, $E_{cl}(k)$, through which the disturbance input $\bar{w}(k)$ enters the system (4.44) depends on the unknown input matrix $B(k)$. Hence, (4.44) cannot be represented using (4.33)-(4.35) alone. To address this, an additional data-driven representation of $B(k)$ is introduced. Supposing (4.32) holds, $B(k)$ can be written as

$$B(k) = \begin{bmatrix} A(k) & B(k) \end{bmatrix} \begin{bmatrix} 0 \\ I_m \end{bmatrix} = \Big(Z(k+1) - W(k)\Big) M(k), \tag{4.50}$$

with $M(k) \in \mathbb{R}^{L\times m}$ satisfying

$$\begin{bmatrix} 0 \\ I_m \end{bmatrix} = \begin{bmatrix} Z(k) \\ U(k) \end{bmatrix} M(k), \tag{4.51}$$

for $k = 0, \dots, T-1$. Using (4.33)-(4.35) and (4.50)-(4.51) the system (4.44) can equivalently be written as a data-dependent lower linear fractional transformation (LFT), see e.g. [205], namely

$$\begin{bmatrix} x(k+1) \\ z(k) \\ \tilde{z}(k) \end{bmatrix} = \begin{bmatrix} Z(k+1)G(k) & \bar{E}_{cl}(k) & I_n \\ C_{cl}(k) & D_{cl}(k) & 0 \\ G(k) & \bar{M}(k) & 0 \end{bmatrix} \begin{bmatrix} x(k) \\ \bar{w}(k) \\ \tilde{w}(k) \end{bmatrix}, \tag{4.52}$$

where $\tilde{w}(k) = -W(k)\tilde{z}(k)$, $\bar{E}_{cl}(k) = [Z(k+1)M(k) \;\; I_n]$ and $\bar{M}(k) = [M(k) \;\; 0]$, together with (4.44c), (4.44d), for $k = 0, \dots, T-1$. Using this data-dependent system representation and the result of Lemma 4.1.2, controllers ensuring the criterion (4.45) holds can be designed directly using noisy data.

Theorem 4.1.4. *Consider the system* (4.1) *and suppose an ensemble of input-output data is available to form the matrices* (4.9), (4.7b)*, such that the rank condition* (4.32) *holds, for* $k = 0, \dots, T-1$. *Suppose* $W(k)$*, as defined in* (4.35)*, satisfies* (4.38)*, for* $k = 0, \dots, T-1$. *Any sequences of matrices* $Y(k)$, $\mathcal{P}(k) = \mathcal{P}(k)^\top$ *satisfying*

$$\begin{bmatrix} \mathcal{P}(k+1) - Q_r(k) & \star & \star & \star & \star \\ -\tilde{S}_p(k)^\top \bar{E}_{cl}(k)^\top & -D_{cl}(k)\tilde{S}_p(k) - \tilde{S}_p(k)^\top D_{cl}(k)^\top + \tilde{R}_p(k) & \star & \star & \star \\ S_r(k)^\top & -\bar{M}(k)\tilde{S}_p(k) & -R_r(k) & \star & \star \\ \bar{E}_{cl}(k)^\top & D_{cl}(k)^\top & \bar{M}(k)^\top & -\tilde{Q}_p(k)^{-1} & \star \\ (Z(k+1)Y(k))^\top & (C(k)\mathcal{P}(k) + D_u(k)U(k)Y(k))^\top & Y(k)^\top & 0 & \mathcal{P}(k) \end{bmatrix} \succ 0, \tag{4.53a}$$

$$Z(k)Y(k) = \mathcal{P}(k), \tag{4.53b}$$

$$\begin{bmatrix} 0 \\ I_m \end{bmatrix} = \begin{bmatrix} Z(k) \\ U(k) \end{bmatrix} M(k), \tag{4.53c}$$

for $k = 0, \dots, T-1$*, and* (4.47b)*, are such that the trajectories of the system* (4.44)*, with*

$$K(k) = U(k)Y(k)\mathcal{P}(k)^{-1}, \tag{4.54}$$

for $k = 0, \dots, T-1$*, and with initial condition* $x(0) = 0$*, satisfy the quadratic robust performance criterion* (4.45).

Proof. By Lemma 4.1.2, (4.45) is satisfied for trajectories of (4.44) if there exists $\mathcal{P}(k) = \mathcal{P}(k)^\top \succ 0$ such that (4.47) holds, for $k = 0, \dots, T-1$. Consider the data-driven system representation (4.52) (based on (4.33)-(4.35) and (4.50)-(4.51)) and let $Y(k) := G(k)\mathcal{P}(k)$. Via the concrete version of the full block S-procedure (see [201], [202]), (4.47a) is satisfied if (4.38) holds and $\mathcal{P}(k)$, $Y(k)$ satisfy

$$\left[\begin{array}{ccc} \star & \star & \star \\ \star & \star & \star \\ \hline \star & \star & \star \\ \star & \star & \star \\ \hline \star & \star & \star \\ \star & \star & \star \end{array}\right]^\top \left[\begin{array}{cc|cc|cc} -\mathcal{P}(k) & 0 & 0 & 0 & 0 & 0 \\ 0 & \mathcal{P}(k+1) & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \tilde{Q}_p(k) & -\tilde{S}_p(k) & 0 & 0 \\ 0 & 0 & -\tilde{S}_p(k)^\top & \tilde{R}_p(k) & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -Q_r(k) & S_r(k) \\ 0 & 0 & 0 & 0 & S_r(k)^\top & -R_r(k) \end{array}\right]$$

$$\times \left[\begin{array}{ccc} (Z(k+1)G(k))^\top & C_{cl}(k)^\top & G(k)^\top \\ I_n & 0 & 0 \\ \hline \bar{E}_{cl}(k)^\top & D_{cl}(k)^\top & \bar{M}(k)^\top \\ 0 & I_q & 0 \\ \hline I_n & 0 & 0 \\ 0 & 0 & I_L \end{array}\right] \succ 0. \quad (4.55)$$

This quadratic form can be transformed into the LMI (4.53a) by performing the matrix multiplication and applying the Schur complement twice. The equality constraints (4.53b) and (4.53c) stem from the upper row block of (4.34) and (4.51), respectively, while the lower row block of (4.34) is automatically satisfied by $K(k)$ in (4.54). □

Theorem 4.1.4 provides a general approach to design controllers guaranteeing robust quadratic performance for unknown LTV systems, affected by both measurement and process noise, directly using noisy data.

Remark 4.1.6. While (4.50)-(4.51) correspond to uniquely identifying the matrix $B(k)$, since the sequence $M(k)$ is determined *at the same time* as the control gain $K(k)$ in (4.54), the result of Theorem 4.1.4 is still a direct data-driven control approach (as opposed to indirect approaches involving *sequential* system identification and control design).

Remark 4.1.7. In the absence of measurement noise, i.e. if $v(k) = 0$, for $k = 0, \ldots, T$, (4.38) reduces to a bound on $D(k)$ (as discussed in Remark 4.1.3). In this case, the closed-loop system is described by (4.44a)-(4.44c) with the disturbance defined as $\bar{w}(k) := d(k)$ and hence $E_{cl}(k) = I_n$ and $D_{cl}(k) = D_d(k)$. This removes the need to represent $B(k)$ via (4.50)-(4.51). The closed-loop system can be represented directly using (4.33)-(4.35) via the LFT (4.52) with $\bar{E}_{cl}(k) = I_n$ and $\bar{M}(k) = 0$ and the data-dependent feasibility problem in Theorem 4.1.4 reduces to finding sequences of matrices $Y(k)$ and $\mathcal{P}(k) = \mathcal{P}(k)^\top$ satisfying (4.53a)-(4.53b), for $k = 0, \ldots, T-1$, and (4.47b). The control law guaranteeing (4.45) for the system (4.44) is given by (4.5), with $K(k)$ given by (4.54).

Remark 4.1.8. While the quadratic robust performance criterion (4.45) considered in this section captures many commonly considered robust performance measures for systems influenced by noise, such as H_∞-control, optimal control in the sense of the results presented in Section 4.1.2 for the noise-free system (4.4) is not considered in the context of noisy data. Designing optimal controllers for unknown systems from noisy data is generally challenging and many open problems remain. In the LTI setting, methods for designing suboptimal LQR controllers or controllers guaranteeing bounded H_2-norm from noisy data are suggested e.g. in [156, 148]. Note that these results focus on systems and data affected by process noise only, and the incorporation of measurement noise as considered in this section is not straightforward. Moreover, in the LTI case, the H_2-norm is well defined. In fact, there are three definitions which are well known to coincide: the 2-norm of the transfer function from the disturbance input to the performance output, the 2-norm of the performance output to unit impulses in the disturbance input channels, and the mean-square deviation of the performance output when the disturbance input is a white process with unit covariance (which is the classic stochastic LQR formulation). However, in the LTV case these definitions do not necessarily coincide, hence, designing H_2-controllers is more involved [206], see e.g. [196, Section 3.5] for H_2-control theory for discrete-time LTV systems. In [207] a finite-horizon performance criterion for LTV systems affected by noise is introduced, which is related to the generalised H_2-norm. Namely, the maximal performance output deviation in response to nonzero initial conditions and disturbance inputs, which considering system (4.1) with $v(k) = 0$ and performance output (4.43) with $D_d(k) = 0$, for $k = 0, \dots, T-1$, and $D_u(T) = 0$, is defined as

$$J_{H_2} = \max_{x(0),d(0),\dots,d(T-1)} \frac{|z|_T}{\left(\|x(0)\|^2 + \sum_{k=0}^{T-1} \|d(k)\|^2\right)^{\frac{1}{2}}}. \tag{4.56}$$

The problem of designing state-feedback controllers of the form (4.5) which minimise (4.56) can be formulated as an SDP [207]. With the choice

$$C(k) = \begin{bmatrix} Q(k)^{1/2} \\ 0 \end{bmatrix}, \; D_u(k) = \begin{bmatrix} 0 \\ R(k)^{1/2} \end{bmatrix},$$

for $k = 0, \dots, T-1$, and $C(T) = Q_f^{1/2}$, the SDP characterising solutions is equivalent to (4.29), i.e. the SDP formulation of the finite-horizon LQR problem for noise-free LTV systems. Suboptimal controllers can be designed directly using noisy data via the data-driven system representation (4.33)-(4.35) by following a similar certainty-equivalence or soft constraint approach as introduced in [156] for LTI systems. Alternatively, in line with the results in this section, supposing the matrix containing unmeasured process noise samples $D(k)$ satisfies (4.38) (with $W(k) = D(k)$), the constraint (4.29d) can be replaced with (4.31d) as in the noise-free case, and the constraint (4.29c) can be replaced with

(4.39) (with $Z(k) = X(k)$, $P(k) = S(k)$ and $Y(k) = H(k)$). While characterising the suboptimality gap of the performance of the data-driven solution obtained via the latter approach with respect to the nominal solution obtained using model knowledge or noise-free data is not straightforward, it depends on the conservatism of the uncertainty bound (4.38), as well as the conservatism of the full block S-procedure (see [201], [202]), which ensures (4.39) and (4.38) imply (4.29c).

4.1.4 Overcoming challenges associated with large time horizons

The data-driven system representations introduced in Corollaries 4.1.1 and 4.1.2 rely on data obtained by exciting the system dynamics via *open-loop* exploring inputs. Due to the time-varying nature of the system dynamics (4.4) or (4.1) the data sequences are required (differently from the LTI case) to cover the *entire* time horizon of interest. For unstable systems with high divergence rates this may lead to numerical issues when solving the feasibility or optimisation problems in Theorems 4.1.1, 4.1.2, 4.1.3 and 4.1.4, if the desired control tasks cover large time horizons. If a controller $\hat{u}_{\hat{K}}(k) = \hat{K}(k)x_d(k)$, which is not necessarily optimal with regards to the design objective, but prevents the closed-loop trajectories from diverging rapidly, is known, experiments can be performed on the closed-loop system by superimposing a sufficiently informative signal $\hat{u}_d(k)$ to ensure the rank condition (4.13) or (4.32) is satisfied with respect to the input $u_d(k) = \hat{u}_{\hat{K}}(k) + \hat{u}_d(k)$, for $k = 0, \ldots, T-1$. If such a controller is not available, or the task is to design such a controller using Theorem 4.1.1 or 4.1.3, an alternative approach to represent the unknown LTV system using data is needed to prevent numerical issues.

Note that similar numerical issues are also encountered in the LTI case, see [95], for which the finite data sequence needs to be sufficiently long. A possible solution in the LTI case, which involves multiple short data sequences in place of a single (longer) data sequence for the data representation and the notion of "collective persistency of excitation" is provided in [119]. However, since the data ensembles in the *time-varying* case are required to capture the time-variation over the entire time horizon of interest, such a strategy is not viable when considering LTV systems. The issue can instead be overcome by collecting successive sets of data sequences covering different subintervals of the considered time horizon. Namely, let the time interval of interest, $k = 0, \ldots, T$, be split into $N_i \in \mathbb{N}$ subintervals

$$\mathcal{I}_i = \{T_{i-1}, \ldots, T_i\} \tag{4.57}$$

for $i = 1, \ldots, N_i$, with $T_0 = 0$ and $T_{N_i} = T$. Assume an ensemble of L data sequences can be collected for each subinterval, $i = 1, \ldots, N_i$. To streamline the presentation consider the noise-free LTV system (4.4) and consider the data matrices

$$X_i(k) = \begin{bmatrix} x_{d,i,1}(k) & x_{d,i,2}(k) & \ldots & x_{d,i,L}(k) \end{bmatrix}, \tag{4.58a}$$

for $k = T_{i-1}, \ldots, T_i$, and

$$U_i(k) = \begin{bmatrix} u_{d,i,1}(k) & u_{d,i,2}(k) & \ldots & u_{d,i,L}(k) \end{bmatrix}, \tag{4.58b}$$

for $k = T_{i-1}, \ldots, T_i - 1$, which are utilised in the following result to provide a system representation based on data from *successive* ensembles of experiments.

Corollary 4.1.3. *Suppose the rank condition*

$$\operatorname{rank}\left(\begin{bmatrix} X_i(k) \\ U_i(k) \end{bmatrix}\right) = n + m, \tag{4.59}$$

holds for $k = T_{i-1}, \ldots, T_i - 1$*,* $i = 1, \ldots, N_i$*,* $T_0 = 0$*, for data gathered during* N_i *ensembles of experiments for the system* (4.4)*. Then, the closed-loop system* (4.6) *can equivalently be represented as*

$$x(k+1) = X_i(k)G(k)x(k), \tag{4.60}$$

where $G(k)$ *satisfies*

$$\begin{bmatrix} I_n \\ K(k) \end{bmatrix} = \begin{bmatrix} X_i(k) \\ U_i(k) \end{bmatrix} G(k), \tag{4.61}$$

for $k = T_{i-1}, \ldots, T_i - 1$*,* $i = 1, \ldots, N_i$*, with* $T_0 = 0$*.*

Proof. Consider the data matrices $X_a(k)$, $U_a(k)$ and $X_b(k)$, $U_b(k)$ obtained from two different ensembles (a and b) of open-loop experiments for the same system (4.4) for $k = 0, \ldots, T$, and satisfying (4.13) for $k = 0, \ldots, T-1$. Using the result of Corollary 4.1.1 the closed-loop system dynamics (4.6) can equivalently be represented in terms of either of the two data matrix sequences, i.e.

$$x(k+1) = X_a(k+1)G_a(k)x(k) = X_b(k+1)G_b(k)x(k), \tag{4.62}$$

where $G_a(k) \in \mathbb{R}^{L \times n}$ and $G_b(k) \in \mathbb{R}^{L \times n}$ satisfy

$$\begin{bmatrix} I_n \\ K(k) \end{bmatrix} = \begin{bmatrix} X_a(k) \\ U_a(k) \end{bmatrix} G_a(k) = \begin{bmatrix} X_b(k) \\ U_b(k) \end{bmatrix} G_b(k),$$

for $k = 0, \ldots, T-1$. Alternatively, the closed-loop system (4.6) can be represented using data from ensemble a for $k = 0, \ldots, T_a - 1$, for any $T_a < T$, and data from ensemble b for the remaining time instances $k = T_a, \ldots, T-1$, i.e. as $x(k+1) = X_i(k+1)G(k)$, for $k = 0, \ldots, T-1$, with $i = a$ and $G(k) = G_a(k)$ for $k = 0, \ldots, T_a - 1$ and $i = b$ and $G(k) = G_b(k)$ for $k = T_a, \ldots, T-1$. Using the same argument, the time horizon of interest can be split into multiple subintervals. In this case the rank condition on the data matrices is only required to hold for the corresponding subinterval, leading to

(4.59). The choice of intervals $\mathcal{I}_i$ as in (4.57) and data matrices (4.58a) and (4.58b), for $i = 1, \ldots, N_i$, with $T_0 = 0$, yield (4.60) - (4.61) for $k = T_{i-1}, \ldots, T_i - 1$, $i = 1, \ldots, N_i$, and $T_0 = 0$. □

Utilising Corollary 4.1.3 instead of Corollary 4.1.1, the control gain sequence and the closed-loop dynamics under state-feedback can be represented using data for control design for bounded closed-loop trajectories or optimal LQR performance via alternative versions of Theorems 4.1.1 and 4.1.2. Following analogous steps as in Corollaries 4.1.3 and 4.1.2 a similar system representation via successive ensembles of data from the noisy LTV system (4.1) can be derived, which allows to design controllers for robust trajectory boundedness and robust performance directly using data via modified versions of Theorems 4.1.3 and 4.1.4.

If the nature of the considered system allows to start experiments partway through the time-varying behaviour we would like to capture, i.e. at $k = \bar{k} > 0$, open-loop data for the matrices $X_i(k)$ or $Z_i(k)$ and $U_i(k)$ can be collected by performing an ensemble of open-loop experiments for $k = T_{i-1}, \ldots, T_i$. However, in many practical situations, this may not be possible. In this case, the successive ensembles of data can be collected in a receding horizon fashion. Namely, data for the first interval $\mathcal{I}_1$, $X_1(k)$ or $Z_1(k)$ and $U_1(k)$, is collected from a set of L open-loop experiments. This data is then used to derive a state-feedback control gain sequence $K_1(k)$ which ensures that (4.19) holds for $k = 0, \ldots, T_1 - 1$ using the result of Theorem 4.1.1 (in the noise-free case), or (4.36), for $k = 0, \ldots, T_1 - 1$ using the result of Theorem 4.1.3 (for systems affected by noise). The data ensemble for the following subinterval $\mathcal{I}_2$ is collected by applying feedback inputs for the time steps covered by the previous interval, $k = 0, \ldots, T_1 - 2$, to ensure that the state-trajectories remain bounded, and open-loop exploring inputs $u_{\exp,2,j}(k)$, $j = 1, \ldots, L$, for $k = T_1 - 1, \ldots, T_2 - 1$, to excite the dynamics and gather sufficiently informative data, $X_2(k)$ or $Z_2(k)$ and $U_2(k)$, for the system representation in the second interval. This data allows to determine a feedback gain sequence ensuring the trajectories remain bounded for $k = 0, \ldots, T_2 - 1$, via modified versions of Theorems 4.1.1 or 4.1.3 based on the alternative system representation using successive ensembles of data as in Corollary 4.1.3. The procedure is then repeated for the following subintervals $\mathcal{I}_i$, $i = 3, \ldots, N_i$. Hence, using this approach the input for each data collection experiment $j = 1, \ldots, L$ in each successive ensemble $i = 1, \ldots, N_i$ is chosen as[8]

$$u_{d,i,j}(k) = \begin{cases} K_{i-1}(k) x_{d,i,j}(k)\,, & \text{for } k = 0, \ldots, T_{i-1} - 2\,, \\ u_{\exp,i,j}(k)\,, & \text{for } k = T_{i-1} - 1, \ldots, T_i - 1\,, \end{cases} \tag{4.63}$$

with $T_0 = 0$ and $T_{N_i} = T$.

Note that in the limit as $T \to \infty$ Theorem 4.1.1 (Theorem 4.1.3) gives a characterisation of stabilising

[8]The exploring inputs start at $k = T_{i-1} - 1$, one time step before the start of interval $\mathcal{I}_i$, to excite the state response and hence ensure different "initial conditions" for the data collection interval at $k = T_{i-1}$ for each experiment in the ensemble. The motivation for this is as detailed in Remark 4.1.2.

(input-to-state stabilising) feedback controllers. For linear arbitrarily time-varying systems, designing controllers with infinite-horizon guarantees, such as stability, is generally a problem involving infinitely many decision variables (both in the model-based and in the data-driven setting). Consider the receding horizon approach described above to collect successive ensembles of data. Rather than repeatedly solving alternative versions of Theorem 4.1.1 or 4.1.3 for a feedback gain sequence for $k = 0, \ldots, T_i - 1$ after collecting data for subinterval $\mathcal{I}_i$, $i = 1, \ldots, N_i$, the gain sequence obtained after the previous ensemble of experiments can be extended to $k = T_{i-1}, \ldots, T_i - 1$, for $i > 1$, by solving (4.17) or (4.39) with $X(k) = X_i(k)$ or $Z(k) = Z_i(k)$ for $k = T_{i-1}, \ldots, T_i$ with an additional constraint ensuring (4.22) is satisfied across the interval boundary. Namely,

$$\begin{bmatrix} P(T_{i-1}) - I_n & X_{i-1}(T_{i-1})Y(T_{i-1}-1) \\ Y(T_{i-1}-1)^\top X_{i-1}(T_{i-1})^\top & P(T_{i-1}-1) \end{bmatrix} \succeq 0, \tag{4.64}$$

in the noise-free case, and

$$\begin{bmatrix} P(T_{i-1}) - I_n - Q_r(T_{i-1}-1) & S_r(T_{i-1}-1) & Z_{i-1}(T_{i-1})Y(T_{i-1}-1) \\ S_r(T_{i-1}-1)^\top & -R_r(T_{i-1}-1) & Y(T_{i-1}-1) \\ Y(T_{i-1}-1)^\top Z_{i-1}(T_{i-1})^\top & Y(T_{i-1}-1)^\top & P(T_{i-1}-1) \end{bmatrix} \succ 0, \tag{4.65}$$

in the noisy case. This approach has the advantage of reducing the computational complexity of the feasibility problem to be solved after each ensemble of experiments, due to the presence of fewer decision variables with respect to solving the problem for the entire time horizon, $k = 0, \ldots, T_i - 1$, after collecting data for interval $\mathcal{I}_i$. However, the feasible solution sets are subsets of those of problems (4.17) and (4.39). Hence, solving the problem sequentially for different subintervals is more restrictive than solving over the entire time horizon and it might not be possible to find a solution to the problem even though a solution to (4.17) or (4.39) exists. Nevertheless, in the limit as $T \to \infty$ the described approach presents a method to design stabilising or input-to-state stabilising state-feedback controllers for unknown LTV systems, using successively obtained *finite-length* data sets. Despite requiring offline experiments, the presented approach, relying on successive ensembles of experiments to capture time-variations, is a step towards online learning of stabilising controllers for linear arbitrarily time-varying systems. In the following, methods to design controllers with infinite-horizon guarantees from finite-length data are presented for the special case of periodically time-varying systems.

4.1.5 Periodically time-varying systems

Consider the special case in which the LTV system (4.1) is such that the time-variation of the matrices $A(k)$ and $B(k)$ is ϕ-periodic, for some $\phi \in \mathbb{N}$, i.e.

$$A(k+\phi) = A(k),\ B(k+\phi) = B(k), \tag{4.66}$$

for all $k \geq 0$. While the system matrices are assumed to be unknown, the periodic nature and period ϕ of the system may be known a priori.

Data-driven system representation

The periodicity of the time-variation of the unknown system (4.1), (4.66) has two important implications for its data-driven representation. Firstly, recall from Section 4.1.1 that L data sequences capturing the time-varying behaviour of the unknown system over the time horizon of interest are required for the representations introduced in Corollary 4.1.1 and Corollary 4.1.2. If the time-varying behaviour is repeated periodically, a single sufficiently long data sequence capturing L periods, i.e. covering the time interval $k = 0, \dots, \phi L$, can replace L data sequences covering the time interval $k = 0, \dots, \phi$. More precisely, consider the state data $x_{d,1,[0,\phi L]}$ or noisy state data $\zeta_{d,1,[0,\phi L]}$. The matrices $X(k)$ in (4.7a) and $Z(k)$ in (4.9) can be populated by letting

$$x_{d,i}(k) = x_{d,1}(k + (i-1)\phi) \text{ and } \zeta_{d,i}(k) = \zeta_{d,1}(k + (i-1)\phi),$$

for $i = 1, \dots, L$, respectively[9]. Secondly, if the control law (4.5) or (4.2) is such that

$$K(k+\phi) = K(k),$$

then the closed-loop system under state-feedback can be represented using the results of Corollary 4.1.1 or Corollary 4.1.2 *beyond* the time interval $k = 0, \dots, T$, $\phi \leq T \leq \phi L$ over which the data has been collected, since $A_{cl}(k+\phi) = A_{cl}(k)$. Namely, the noise-free closed-loop system (4.6) can equivalently be represented as

$$x(k+1) = X(l+1)G(l)x(k), \tag{4.67}$$

and the controller as

$$u(k) = U(l)G(l)x(k), \tag{4.68}$$

for all $k \geq 0$, with $l = k - n_p\phi$, $n_p = \max\{r \in \mathbb{Z} \mid r \leq k/\phi\}$, if $X(l)$, $U(l)$ satisfy (4.13) and $G(l)$ is such that (4.15) holds (with $k = l$), for $l = 0, \dots, \phi - 1$. Similarly, the dynamics matrix of the

[9] Similarly, the required data can be obtained via $1 \leq N_e \leq L$ experiments jointly covering L periods.

closed-loop system (4.3) can equivalently be represented as

$$A(k) + B(k)K(k) = \Big(Z(l+1) + W(l)\Big)G(l), \tag{4.69}$$

and the controller as

$$u(k) = U(l)G(l)\zeta(k), \tag{4.70}$$

for all $k \geq 0$, with $l = k - n_p\phi$, $n_p = \max\{r \in \mathbb{Z} \mid r \leq k/\phi\}$, if $Z(l)$, $U(l)$ satisfy (4.32) and $G(l)$ is such that (4.34) holds (with $k = l$), for $l = 0, \dots, \phi - 1$.

These observations make it possible to derive *infinite-horizon* results for this class of system based on *finite-horizon* data. In the remainder of this section, the infinite-horizon versions of the control problems considered in Section 4.1.2 and Section 4.1.3 are considered in the context of periodically time-varying systems.

Stabilisation

Consider the system (4.4), (4.66) and the problem of designing state-feedback controllers, which stabilise the closed-loop system (4.6). Exploiting periodicity, this problem can be solved directly using finite input-state data.

Corollary 4.1.4. *Consider the linear periodically time-varying system* (4.4)*,* (4.66) *and suppose input-state data is available to form the matrices* (4.7a)*,* (4.7b)*, such that the rank condition* (4.13) *holds, for* $k = 0, \dots, \phi-1$*. Any sequences of matrices* $Y(k)$*,* $P(k) = P(k)^\top$ *satisfying* (4.17)*, for* $k = 0, \dots, \phi-1$*, where* $\eta \geq 1$ *and* $\rho > \eta$ *are finite constants, and*

$$P(\phi) = P(0), \tag{4.71}$$

are such that the zero equilibrium of the system (4.6)*,* (4.66)*, with* $K(k)$ *given by* (4.18)*, for* $k = 0, \dots, \phi - 1$*, and* $K(k + n_p\phi) = K(k)$*, for all* $n_p \geq 0$*, is exponentially stable.*

Proof. The zero equilibrium of the closed-loop LTV system (4.6) is exponentially stable if and only if there exists $P(k) = P(k)^\top$ satisfying (4.17c) and (4.22) for some $K(k)$ for all $k \geq 0$. If the system dynamics (4.6) are ϕ-periodic the zero equilibrium is exponentially stable if and only if there exists a ϕ-periodic solution $P(k)$, $K(k)$ to (4.22) [196, Section 3.1]. Hence, $K(k)$, $P(k)$ satisfying the recursive inequality (4.22) only need to be determined for one period, i.e. for $k = 0, \dots, \phi$, with the additional boundary condition constraint (4.71) in place to ensure that $P(k)$ is periodic. Using (4.14), (4.15) and following steps similar as in the proof of Theorem 4.1.1, (4.22) is equivalent to (4.17). □

Corollary 4.1.4 is the infinite-horizon equivalent of Theorem 4.1.1 for periodically time-varying systems. Similarly, (noise) input-to-state stabilising controllers can be designed using noisy data for the system (4.1), (4.66). This represents the infinite-horizon counterpart to Theorem 4.1.3.

Corollary 4.1.5. *Consider the linear periodically time-varying system* (4.1), (4.66) *and suppose an ensemble of input-output data is available to form the matrices* (4.9), (4.7b), *such that the rank condition* (4.32) *holds, for* $k = 0, \ldots, \phi - 1$. *Suppose* $W(k)$ *satisfies* (4.38), *for* $k = 0, \ldots, \phi - 1$. *Any sequences of matrices* $Y(k)$, $P(k) = P(k)^\top$ *satisfying* (4.39) *and* (4.17c) *for* $k = 0, \ldots, \phi - 1$, *where* $\eta \geq 1$ *and* $\rho > \eta$ *are finite constants, and* (4.71), *are such that the zero equilibrium of the system* (4.3), (4.66), *with* $K(k)$ *given by* (4.18), *for* $k = 0, \ldots, \phi - 1$, *and* $K(k + n_p\phi) = K(k)$, *for all* $n_p \geq 0$, *is input-to-state stable, with respect to the disturbance inputs* v, d.

Proof. By Theorem 4.1.3 the trajectories of (4.3) with $K(k)$ given by (4.18) satisfy (4.36), for $0, \ldots, \phi - 1$, if there exist $Y(k)$, $P(k) = P(k)^\top$ satisfying (4.39) and (4.17c) for $k = 0, \ldots, \phi - 1$. If (4.66) and $K(k + n_p\phi) = K(k)$, for all $n_p \geq 0$, then (4.36) holds for all $k \geq 0$. Since the first term on the right hand side of (4.36) tends to zero as $k \to \infty$, this corresponds to input-to-state stability. □

Optimal control

Consider the system (4.4), and the problem of finding a stabilising $u^\star(k)$, for all $k \geq 0$, minimising

$$J\left(x(0), u(\cdot)\right) = \sum_{k=0}^{\infty} \left(x(k)^\top Q(k) x(k) + u(k)^\top R(k) u(k) \right), \tag{4.72}$$

with $Q(k) = Q(k)^\top \succeq 0$ and $R(k) = R(k)^\top \succ 0$, for all $k \geq 0$. If (4.66) holds and $Q(k+\phi) = Q(k)$ and $R(k + \phi) = R(k)$, then the sequence of state-feedback gains $K^\star(k)$, $k \geq 0$, corresponding to the solution $u^\star(k)$ as given by (4.26) is also ϕ-periodic, i.e. $K^\star(k + \phi) = K^\star(k)$ [196, Section 3.1]. Similarly to the finite-horizon case considered in Section 4.1.2, the described *infinite-horizon* LQR problem can be formulated and solved via a convex programme involving LMI constraints [198]. Exploiting periodicity, this problem can be solved directly using finite input-state data.

Corollary 4.1.6. *Consider the linear periodically time-varying system* (4.4), (4.66) *and suppose input-state data is available to form the matrices* (4.7a), (4.7b), *such that the rank condition* (4.13) *holds, for* $k = 0, \ldots, \phi - 1$. *Consider the cost function* (4.72) *with* $Q(k+\phi) = Q(k)$ *and* $R(k+\phi) = R(k)$, *for all* $k \geq 0$. *The optimal state-feedback control gain sequence solving the infinite-horizon LQR problem with* $u^\star(k) = K^\star(k)x(k)$ *is given by* (4.30), *for* $k = 0, \ldots, \phi - 1$, *and* $K^\star(k + n_p\phi) = K^\star(k)$, *for all*

$n_p \geq 0$, with $H^\star(k)$ and $S^\star(k)$ the solution of

$$\begin{aligned} \min_{\mathcal{S},\mathcal{H},\mathcal{O}} \quad & \sum_{k=0}^{\phi-1} \Big(\mathbf{Tr}\left(Q(k)S(k)\right) + \mathbf{Tr}\left(O(k)\right) \Big) \\ \text{s.t.} \quad & (4.31\text{b}) - (4.31\text{e}), \\ & S(\phi) = S(0), \end{aligned} \tag{4.73}$$

for $k = 0, \dots, \phi - 1$, where

$$\mathcal{S} = \{S(1), \dots, S(\phi)\}, \; \mathcal{H} = \{H(0), \dots, H(\phi-1)\} \; \textit{and} \; \mathcal{O} = \{O(0), \dots, O(\phi-1)\}.$$

Proof. The infinite-horizon LQR problem can be recast as a convex programme (see [198]). Then, exploiting that the solution is a state-feedback law and introducing (4.14), (4.15) yields (4.31) with $Q_f = 0$ and $T \to \infty$, where $H(k) := G(k)S(k)$. Recall that $K^\star(k)$ for the considered problem is ϕ-periodic [196, Section 3.1]. It remains to be shown that this ϕ-periodic solution can be recovered by solving (4.73) over one period, with the additional constraint $S(\phi) = S(0)$. Since $K^\star(k)$ is stabilising by construction, there exists a ϕ-periodic solution $S^\star(k+\phi) = S^\star(k)$ satisfying (4.31b) and (4.31c) (this can be shown using analogous arguments as in the proof of Corollary 4.1.4). Thus, the solution of the slack variable $O^\star(k) = R(k)^{\frac{1}{2}} K^\star(k) S^\star(k) K^\star(k)^\top R(k)^{\frac{1}{2}}$ is also ϕ-periodic. Hence, the constraints (4.31b) - (4.31e) are satisfied at time $k+n_p\phi$, for all $n_p \geq 0$, if they are satisfied at time k. Similarly, the optimal stage cost $I_c^\star(k) = \mathbf{Tr}\left(Q(k)S^\star(k)\right) + \mathbf{Tr}\left(O^\star(k)\right)$, satisfies $I_c^\star(k+n_p\phi) = I_c^\star(k)$, for all $n_p \geq 0$. Hence, the optimal cost is given by $\sum_{k=0}^{\infty} I_c^\star(k) = \lim_{n_p\to\infty} n_p \sum_{k=0}^{\phi-1} I_c^\star(k)$. Note that $\sum_{k=0}^{\phi-1} I_c^\star(k)$ is the optimal cost obtained by solving (4.73). Hence, the periodic solution to the infinite-horizon LQR problem is given by $K^\star(k)$, for $k = 0, \dots, \phi-1$, solving (4.73), and $K^\star(k+n_p\phi) = K^\star(k)$, for all $n_p \geq 0$. □

Robust performance

Consider the problem of designing stabilising controllers of the form (4.2), such that the closed-loop system (4.44a)-(4.44b), (4.44d) satisfies the infinite-horizon performance criterion

$$\sum_{k=0}^{\infty} \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix}^\top \begin{bmatrix} Q_p(k) & S_p(k) \\ S_p(k)^\top & R_p(k) \end{bmatrix} \begin{bmatrix} \bar{w}(k) \\ z(k) \end{bmatrix} + \varepsilon \sum_{k=0}^{\infty} \bar{w}(k)^\top \bar{w}(k) \leq 0, \tag{4.74}$$

for all $\bar{w} \in \ell_2$, with $\varepsilon > 0$, $R_p(k) \succeq 0$. As in Section 4.1.3, assume the performance index matrix is invertible and the inverse is given by (4.46), such that $\tilde{Q}_p(k) \prec 0$, for all $k \geq 0$. Suppose the system dynamics and the performance index are ϕ-periodic, i.e. (4.66) holds and

$$\begin{aligned} C(k+\phi) &= C(k), & Q_p(k+\phi) &= Q_p(k), \\ D_u(k+\phi) &= D_u(k), & S_p(k+\phi) &= S_p(k), \\ D_d(k+\phi) &= D_d(k), & R_p(k+\phi) &= R_p(k). \end{aligned}$$

Exploiting periodicity, this problem can be solved directly using finite input and noisy state data.

Corollary 4.1.7. *Consider the linear periodically time-varying system* (4.1), (4.66) *and suppose input-output data is available to form the matrices* (4.9), (4.7b), *such that the rank condition* (4.32) *holds, for* $k = 0, \dots, \phi - 1$. *Suppose the performance index is* ϕ*-periodic and* $W(k)$*, as defined in* (4.35), *satisfies* (4.38), *for* $k = 0, \dots, \phi - 1$. *Any sequences of matrices* $Y(k)$, $\mathcal{P}(k) = \mathcal{P}(k)^\top$ *satisfying* (4.53a)-(4.53c), *for* $k = 0, \dots, \phi - 1$, *and*

$$\mathcal{P}(\phi) = \mathcal{P}(0), \tag{4.75}$$

are such that the trajectories of the system (4.44a), (4.44b), (4.44d), *with* $K(k)$ *given by* (4.54), *for* $k = 0, \dots, \phi - 1$, *and* $K(k + n_p\phi) = K(k)$, *for* $n_p \geq 0$, *and with initial condition* $x(0) = 0$, *satisfy the quadratic robust performance criterion* (4.74).

Proof. Analogous to Lemma 4.1.2 it can be shown via dissipativity arguments (see e.g. [204]) and the dualisation lemma [201, Lemma 4.9] that (4.74) holds, if there exist ϕ-periodic sequences $K(k)$, $\mathcal{P}(k) = \mathcal{P}(k)^\top$ satisfying (4.47a) for all $k \geq 0$. Stability is implied by the upper left block of (4.47a) and the assumption that $\tilde{Q}_p(k) \prec 0$ for all $k \geq 0$. The data-driven formulation (4.53), (4.75) follows via analogous steps to those in the proof of Theorem 4.1.4, exploiting periodicity. □

4.1.6 Examples

The efficacy and the practical relevance of the results presented in Sections 4.1.2 - 4.1.5 are demonstrated via the following examples. Firstly, two numerical simulation examples illustrate the efficacy of the results for control of noise-free LTV systems over short and long time horizons. Secondly, the results for periodically time-varying systems, both in the noise-free case and in the case in which the system is affected by noise, are demonstrated via a practically motivated simulation example involving a voltage source converter. Finally, the benefits of the results for robust data-driven control of arbitrarily varying LTV systems for robotics applications are highlighted by considering the problem of controlling a snake-like robot.

Illustrative numerical examples

The results for noise-free LTV systems in Section 4.1.2 and the challenges associated with designing data-driven controllers for linear arbitrarily time-varying systems over large time horizons discussed in Section 4.1.4 are illustrated via the following two numerical examples.

Finite-horizon LQR: Consider the LTV system (4.4) with

$$A(k) = \begin{bmatrix} 1 & 0.0025k \\ -0.1\cos(0.3k) & 1 + 0.05^{3/2}\sin(0.5k)\sqrt{k} \end{bmatrix},$$

$$B(k) = 0.05 \begin{bmatrix} 1 \\ \frac{0.1k+2}{0.1k+3} \end{bmatrix},$$

as introduced in [193]. Note that the system is open-loop unstable and consider the problem of finding a feedback gain sequence $K^\star(k)$ which minimises the cost function (4.25), with

$$Q(k) = (0.04k + 2)\, I_2,\ R(k) = 5 - 0.02k,\ Q_f = 50I_2,$$

over the time horizon $k = 0, \dots, T-1$, with $T = 120$. The data for the data-driven system representation is gathered in $L = 3$ open-loop simulations with initial conditions sampled from a uniform distribution on the interval $(0, 1)$ and by applying an input sequence also sampled randomly from a uniform distribution on $(0, 1)$ over the time interval $[0, T-1]$. The data-dependent optimisation problem (4.31) is solved using CVX [208]. For comparison, the optimal solution is also computed by solving (4.28) (using model knowledge). The sequence of control gains $K^\star(k)$ computed using the data-based representation (i.e. the result given in Theorem 4.1.2) coincides with the control sequence $\bar{K}(k)$, for $k = 0, \dots, T-1$, obtained by recursively solving the difference Riccati equation (4.28) with an average error $\|K^\star(k) - \bar{K}(k)\|$ of order 10^{-8}. The time histories of the first (top plot) and second (bottom plot) components of the state of the closed-loop system with $\{K^\star(0), \dots, K^\star(T-1)\}$ and $x_0 = [0.4411\ 0.2711]^\top$ are shown in Figure 4.1. The corresponding input sequence (top plot) and the gain error $\|K^\star(k) - \bar{K}(k)\|$ (bottom plot) for each time instance are shown in Figure 4.2.

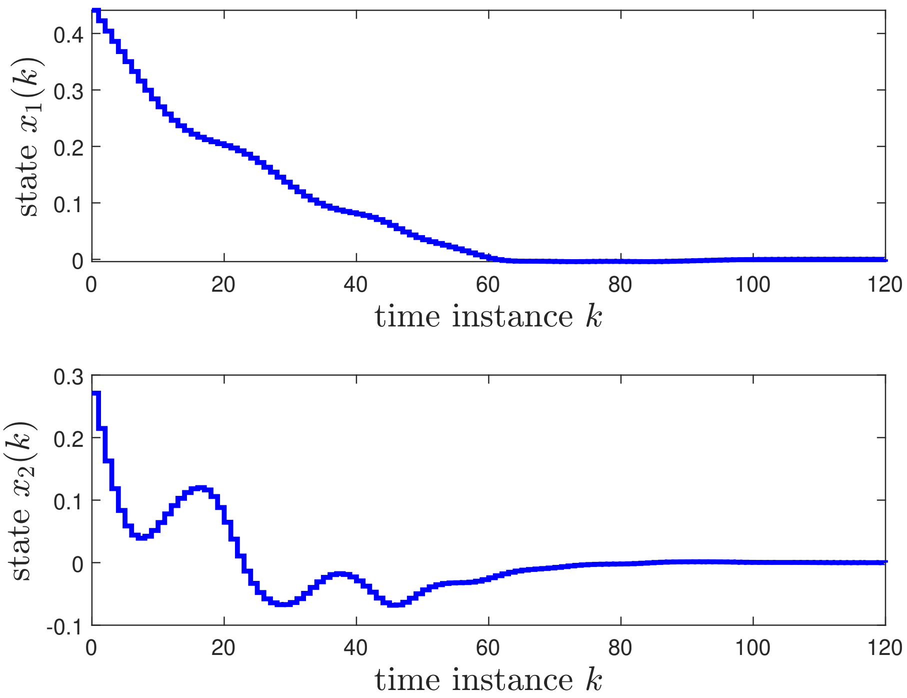


Figure 4.1: The time histories of the states of the system in closed-loop with the optimal gain sequence determined from (4.31). (© 2020 IEEE)

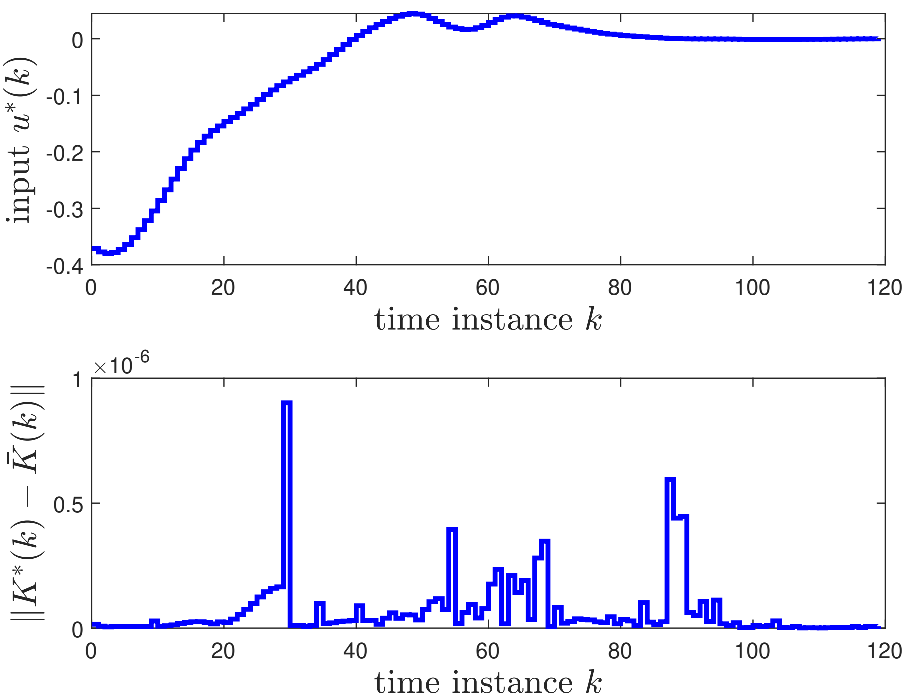


Figure 4.2: The time histories of the optimal input sequence $u^\star(k) = K^\star(k)x(k)$ of the system (top) and the error between the optimal control gains (bottom) determined from (4.31) (model-free) and (4.28) (model-based). (© 2020 IEEE)

Bounded closed-loop trajectories over a larger time horizon: Consider the scalar LTV system

$$x(k+1) = (1.3 + 0.08\sin(0.05k)\cos(0.1k))\,x(k) + \ 0.2\cos(0.2k)u(k), \tag{4.76}$$

which is open-loop unstable. Consider the problem of determining a control gain sequence ensuring (4.19) holds, for $k = 0, \ldots, T$, with $T = 150$. In an attempt to solve the problem via Theorem 4.1.1, the input-state data for an ensemble of $L = 2$ experiments covering the entire time horizon, starting from initial conditions randomly sampled from a uniform distribution on the interval $(0, 1)$, is shown in Figure 4.3. The state trajectories diverge rapidly resulting in that (4.17) cannot be solved successfully. Considering instead the receding horizon approach outlined in Section 4.1.4 to collect $N_i = 3$ *successive* data ensembles of $L = 2$ experiments, with $T_1 = 50$, $T_2 = 100$ and $T_3 = T = 150$, input-state data collected over the same time horizon is shown in Figure 4.4. The data is arranged to form the matrices $X_i(k)$, $U_i(k)$, $i = 1, 2, 3$. Using these matrices for the data-driven system representation based on successive ensembles of data described in Corollary 4.1.3, instead of the data-driven representation based on a single data ensemble covering the entire time horizon as in Corollary 4.1.1, in the context of Theorem 4.1.1, a gain sequence $K(k)$ ensuring that (4.19) holds, for $k = 0, \ldots, T$, and hence solving the considered control task, is obtained. The corresponding closed-loop response, subject to the initial condition $x_0 = 0.4795$ (blue) as well as the bound (4.19) (red) are illustrated in Figure 4.5.

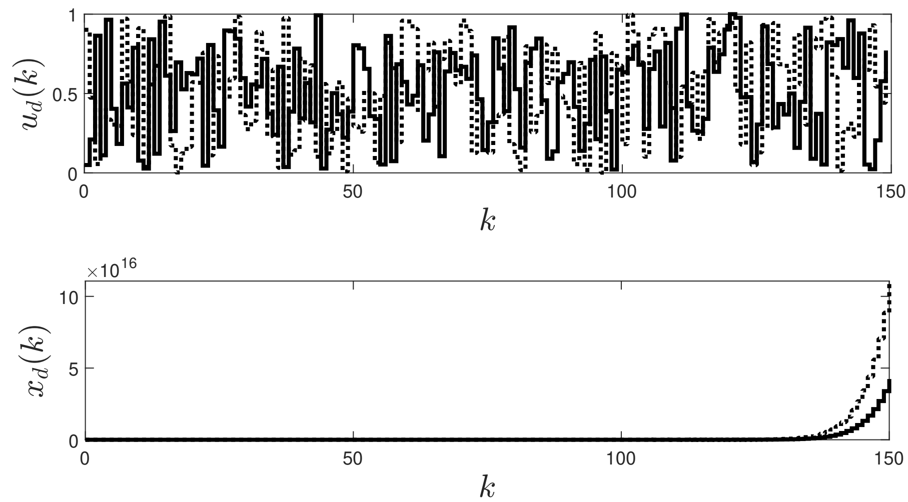


Figure 4.3: Time histories of the input and state of the system (4.76) from an ensemble of two experiments, (indicated by the solid and dotted lines), for $k = 0, \ldots, T$, starting from initial conditions randomly sampled from a uniform distribution on the interval $(0, 1)$.

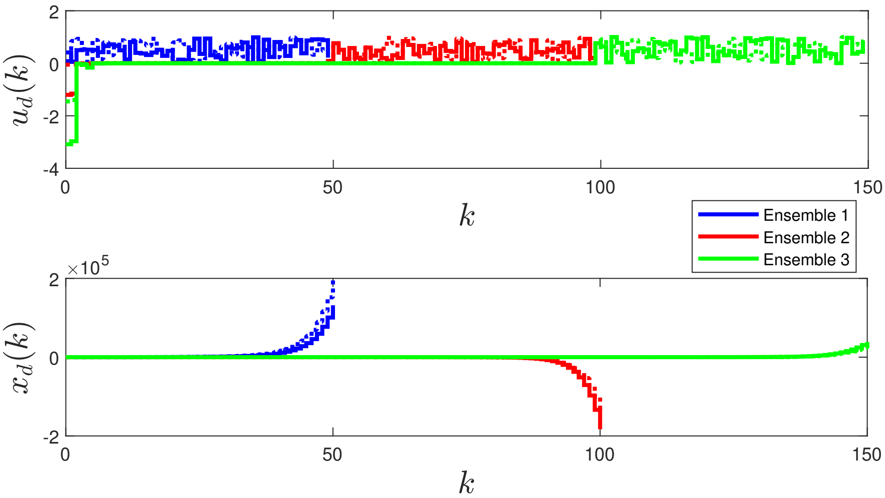


Figure 4.4: Time histories of the input and state of the system (4.76) for three successive ensembles (covering intervals up to T_1, T_2 and T_3 indicated in blue, red and green, respectively) of two experiments (indicated by the solid and dotted lines), starting from initial conditions randomly sampled from a uniform distribution on the interval $(0, 1)$.

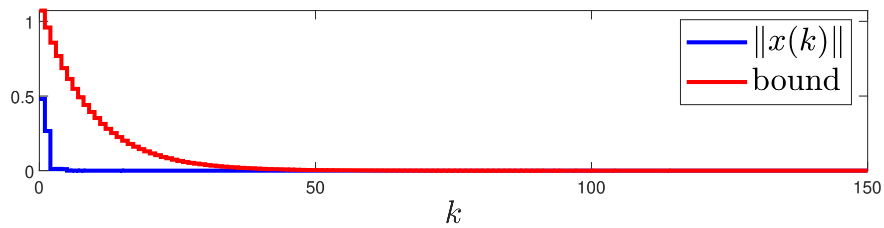


Figure 4.5: Time history of the state of the closed-loop system (4.76) with $K(k)$ obtained using data from three successive ensembles.

Power converter

The efficacy of the results for periodically time-varying systems in Section 4.1.5 is demonstrated via a practically motivated example. Consider the average equivalent model of a single-phase power converter as described in [209] and depicted in Figure 4.6, i.e. consider the system

$$\frac{d}{dt}\begin{bmatrix} v_{dc} \\ i_{lg} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{R}{L} \end{bmatrix}\begin{bmatrix} v_{dc} \\ i_{lg} \end{bmatrix} + \begin{bmatrix} \frac{-i_{lg}}{Cv_{dc}} \\ \frac{1}{L} \end{bmatrix} v_l + \begin{bmatrix} \frac{1}{C} & 0 \\ 0 & -\frac{1}{L} \end{bmatrix}\begin{bmatrix} i_s \\ v_g \end{bmatrix}, \tag{4.77}$$

where C is the capacitance of the direct current (DC) bus, R and L are the resistance and inductance associated with a grid filter, the DC bus voltage v_{dc} and the alternating current (AC) grid current i_{lg} are the system state variables, the converter voltage v_l is the control input and the current entering the DC bus i_s and the AC grid voltage v_g are exogenous signals, which are considered as disturbances. Let ω denote the grid frequency and consider the parameters given in Table 4.1. Linearising (4.77)

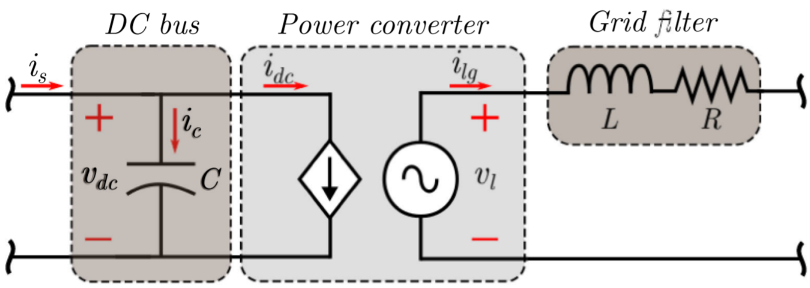


Figure 4.6: Average equivalent model of a single-phase power converter [209].

around the reference trajectory

$$\begin{aligned}
\tilde{v}_{dc}(t) &= V_{DC}, \\
\tilde{i}_{lg}(t) &= \sqrt{2}/V_{g,rms}\left(\tilde{P}_s\cos(\omega t) + \tilde{Q}\sin(\omega t)\right), \\
\tilde{v}_g(t) &= \sqrt{2}V_{g,rms}\cos(\omega t), \\
\tilde{v}_l(t) &= L\frac{d}{dt}\tilde{i}_{lg}(t) + R\tilde{i}_{lg}(t) + \tilde{v}_g(t), \\
\tilde{i}_s(t) &= C\frac{d}{dt}\tilde{v}_{dc}(t) + \tilde{i}_{lg}(t)\tilde{v}_l(t)/\tilde{v}_{dc}(t),
\end{aligned}$$

and discretising using forward Euler with a time step $\Delta = 0.0005s$ results in an open-loop unstable linear periodically time-varying system of the form (4.1), (4.66), with period $\phi = 40$, with $x = [\delta v_{dc}, \delta i_{lg}]^\top$, $u = \delta v_l$ and $d = \operatorname{diag}(\Delta/C, -\Delta/L)[\delta i_s, \delta v_g]^\top + d_h$, where δ indicates deviations from

Table 4.1: Power converter parameters for the simulation example.

Parameter	R	L	C	V_{DC}	ω	$V_{g,rms}$	$\tilde{P}_s$	$\tilde{Q}$
Value	0.06 Ω	0.101 mH	0.89 mF	110 V	50 Hz	50 V	300 W	300 W

the reference trajectory, e.g. $\delta v_{dc} = v_{dc} - \tilde{v}_{dc}$, and d_h contains the higher-order terms of the Taylor series expansion. Considering the LTV dynamics as unknown, the aim is to design feedback controllers that maintain the system (4.77) on the reference trajectory, directly using data. Data for the system representation is gathered via a single open-loop simulation capturing $L = 3$ periods, i.e. for $k = 0, \dots, 120$, with initial condition $x_d(0)$ randomly generated from a uniform distribution on the interval $(0, 0.1)$. The exploring inputs $u_d(k)$, for $k = 0, \dots, 119$, are randomly generated from a uniform distribution on $(0, 0.01)$. The data-dependent optimisation problems are solved using CVX [208].

LQR: Consider the infinite-horizon LQR problem described by the LTV approximation of the power converter (neglecting the higher-order terms and assuming the exogenous signals follow their reference trajectories, i.e. $d = 0$) and the cost function (4.72), with $Q(k) = \mathrm{diag}\left(0.7 + 0.2\cos(\frac{\pi}{5}k + \theta), 0.3 - 0.2\cos(\frac{\pi}{5}k + \theta)\right)$ and $R(k) = 0.001$, for all $k \geq 0$, where $\theta = -4.1278$ rad. Corollary 4.1.6 can be used to determine the optimal gain sequence $K^\star(k)$ directly using data. For comparison, the solution to the LQR problem is also computed using the LTV model to obtain the optimal control gain sequence $\bar{K}^\star(k)$. The optimal gain sequence computed directly using data $K^\star(k)$ coincides with $\bar{K}^\star(k)$ with an average error $\|K^\star(k) - \bar{K}^\star(k)\|$ of order 10^{-8} over one period. The time histories of the first (black) and second (blue) components of the state of the closed-loop system with $K^\star(k)$ and initial condition $x_0 = [1.1236,\ 0]^\top$, which corresponds to a disturbance input

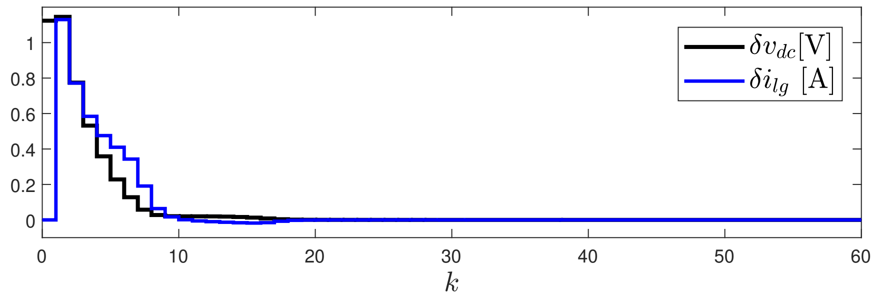


Figure 4.7: The time histories of the states of the power converter system in closed-loop with the optimal gain sequence determined from (4.73).

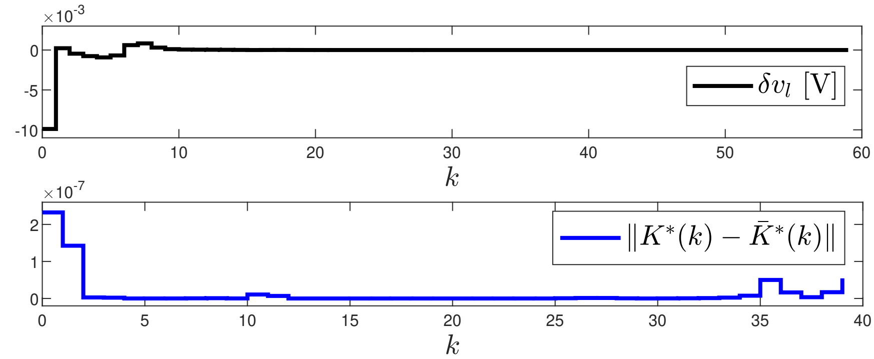


Figure 4.8: The time histories of the optimal input sequence $u^\star(k) = K^\star(k)x(k)$ for the power converter system (top) and the error between the optimal control gains (bottom) determined from (4.73) (data-driven) and (4.28) (model-based) over one period.

$\delta i_s = 2$ A at $k = -1$, are shown in Figure 4.7 for a simulation horizon of 60 time instances. The corresponding input sequence for each time instance (top plot) and the gain error $\|K^\star(k) - \bar{K}^\star(k)\|$ (bottom plot) over one period, i.e. $k = 0, \ldots, \phi - 1$, are shown in Figure 4.8.

$\mathbf{H_\infty}$-control: Consider now the robust performance criterion (4.74) with $Q_p(k) = -\bar{\gamma}^2 I_{(m+n)}$, $\bar{\gamma} > 0$, $S_p(k) = 0$ and $R_p(k) = I_q$, for all $k \geq 0$. Differently from the LQR case, state data for the system representation is generated by simulating the response of the nonlinear system (4.77), rather than the LTV approximation. While $\delta i_s(k) = 0$ and $\delta v_g(k) = 0$ for $k = 0, \ldots, 120$, the higher order terms in the dynamics are regarded as process noise, i.e. $d = d_h$. In addition, measurement noise $v(k)$ is simulated by adding a random signal uniformly sampled on the interval $(-0.001, 0.001)$ to the state samples. Corollary 4.1.7 is used to design a disturbance attenuating controller directly using data. It can be verified that there exists a sequence of matrices $Q_r(k)$, $S_r(k)$ and $R_r(k)$ such that the resulting $W(k)$, for $k = 0, \ldots, 39$ (one period), satisfies (4.38) and such that (4.53a)-(4.53c), (4.75) are feasible. Via a line-search, the lowest bound on the H_∞ norm for which the problem (4.53a)-(4.53c), (4.75) based on noisy data is feasible is found to be $\bar{\gamma} = 7.9$. This is the same value that could be achieved by solving the equivalent model-based problem. To test the data-driven robust controller the response of the voltage source converter (4.77) to step changes in the mean value of the DC bus current, such that $\delta i_s = 1.5$ A for $10 \leq k \leq 79$, is simulated. The time histories of the control input and the corresponding state response of the closed-loop error system, are compared to the case in which the control input follows the reference trajectory without any feedback action, i.e. $\delta v_l(k) = 0$ for all k, in Figure 4.9. In the latter case, the disturbance input in δi_s causes v_{dc} to rise (black). When δv_l is regulated using the data-based H_∞-controller determined by solving (4.53a)-(4.53c), (4.75), v_{dc} remains close to its nominal trajectory by releasing power into the AC grid (blue).

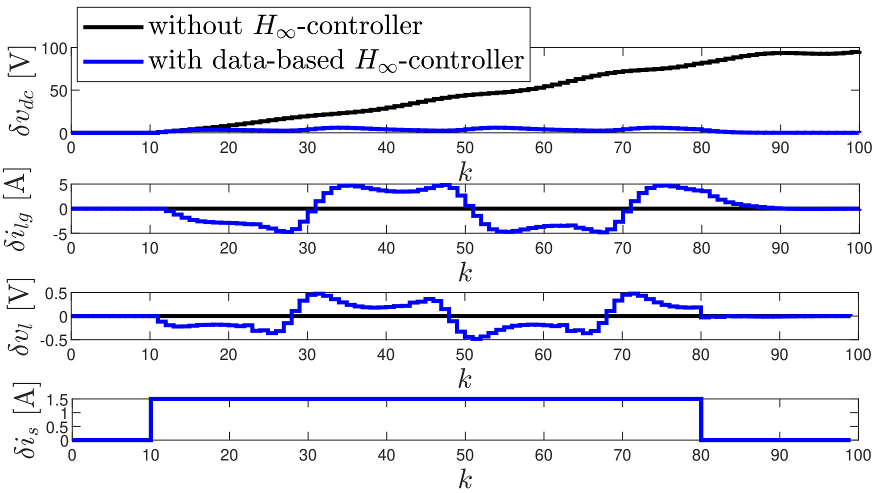
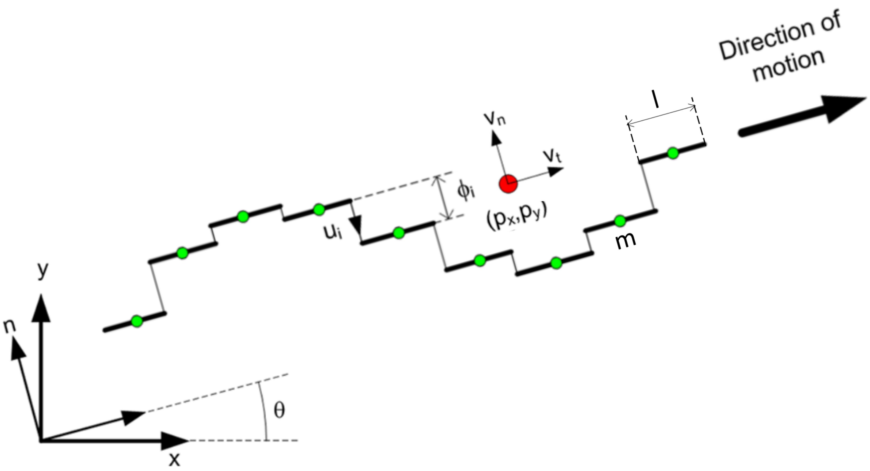


Figure 4.9: Time histories of the state and input response to a step input in the disturbance δi_s with and without the robust controller designed directly using noisy data.

Snake-like robot

The practical relevance of the results in Section 4.1.3 is highlighted via an application example involving locomotion control of a snake-like robot. An example of a snake-like robot is pictured in Figure 4.10a. Mimicking the motion of biological snakes (see e.g. [210, 211]), such robots have gained interest for a wide range of applications, due to their excellent mobility and maneuverability, even in hard-to-reach, challenging environments. However, obtaining accurate mathematical models describing the dynamics of snake-like robots is challenging. A qualitative model of their dynamic behaviour for control design, based on the diagram in Figure 4.10b is presented in [212], [3, Chapter 6]. The system

(a) A photo of the snake-like robot “Kulko” [3].



(b) Simplified model of a snake-like robot (modified from [3]).

Figure 4.10: Snake-like robot.

state is $\xi = \begin{bmatrix}\phi^\top & \theta & p_x & p_y & v_\phi^\top & v_\theta & v_t & v_n\end{bmatrix}^\top \in \mathbb{R}^{2N_l+4}$, where $\phi = \begin{bmatrix}\phi_1 & \dots & \phi_{N_l-1}\end{bmatrix}^\top \in \mathbb{R}^{N_l-1}$ and $v_\phi = \begin{bmatrix}v_{\phi_1} & \dots & v_{\phi_{N_l-1}}\end{bmatrix}^T \in \mathbb{R}^{N_l-1}$ combine the $N_l - 1$ relative link displacements and corresponding velocities, where $N_l \in \mathbb{N}$ denotes the number of links, $\theta \in \mathbb{R}$ denotes the heading of the robot, which corresponds to the angle by which the body (t, n)-coordinate frame is rotated with respect to the global (x, y)-coordinate frame, $p_x \in \mathbb{R}$ and $p_y \in \mathbb{R}$ are the coordinates of the centre of gravity of the robot in the global frame, and $v_\theta \in \mathbb{R}$, $v_t \in \mathbb{R}$ and $v_n \in \mathbb{R}$ are the corresponding velocities representing the rates of change of the heading angle and the centre of gravity position in the body frame, respectively. The control inputs are the actuator forces $u_i \in \mathbb{R}$, $i = 1, \ldots, N_l - 1$, at the joints. Let $\upsilon = \begin{bmatrix} u_1 & \ldots & u_{N_l-1} \end{bmatrix}^\top$, the dynamics of the snake-like robot are then described by (see [212], [3, Chapter 6])

$$\dot{\phi} = v_\phi, \tag{4.78a}$$

$$\dot{\theta} = v_\theta, \tag{4.78b}$$

$$\dot{p}_x = v_t \cos\theta - v_n \sin\theta, \tag{4.78c}$$

$$\dot{p}_y = v_t \sin\theta + v_n \cos\theta, \tag{4.78d}$$

$$\dot{v}_\phi = -\frac{c_n}{m} v_\phi + \frac{c_p}{m} v_t A D^T \phi + \frac{1}{m} D D^T \upsilon, \tag{4.78e}$$

$$\dot{v}_\theta = -\lambda_1 v_\theta + \frac{\lambda_2}{N_l - 1} v_t \bar{e}^T \phi, \tag{4.78f}$$

$$\dot{v}_t = -\frac{c_t}{m} v_t + \frac{2c_p}{N_l m} v_n \bar{e}^T \phi - \frac{c_p}{N_l m} \phi^T A \bar{D} v_\phi, \tag{4.78g}$$

$$\dot{v}_n = -\frac{c_n}{m} v_n + \frac{2c_p}{N_l m} v_t \bar{e}^T \phi\,, \tag{4.78h}$$

where c_n and c_t denote the friction coefficients in the normal and tangential directions, respectively, c_p denotes the propulsion coefficient defined as $c_p = (c_n - c_t)/(2l)$, with m and l denoting the mass and length of a single link, respectively, and where $\bar{e} = \begin{bmatrix} 1 & \ldots & 1 \end{bmatrix}^\top \in \mathbb{R}^{N_l - 1}$, $A \in \mathbb{R}^{(N_l-1)\times N_l}$ and $D \in \mathbb{R}^{(N_l-1)\times N_l}$ denote the matrices

$$A = \begin{bmatrix} 1 & 1 & & \\ & \ddots & \ddots & \\ & & 1 & 1 \end{bmatrix}, \quad D = \begin{bmatrix} 1 & -1 & & \\ & \ddots & \ddots & \\ & & 1 & -1 \end{bmatrix},$$

$\bar{D} = D^T (D D^T)^{-1} \in \mathbb{R}^{N_l \times (N_l - 1)}$, and λ_1, λ_2 are rotational parameters. On planar surfaces, the under-actuated robotic system achieves forward motion via *lateral undulation*, namely the joint displacements track the gait pattern

$$\phi_{i,\mathrm{loc}}(t) = \alpha \sin(\omega t + (i-1)\beta) + \gamma, \tag{4.79}$$

where α, β, ω and γ are parameters which can be used to prescribe a desired trajectory for the centre of gravity of the snake-like robot, see [3] for more details. If the system model is known, controllers ensuring this locomotion reference (4.79), $i = 1, \ldots, N_l - 1$, is tracked can be designed via a partial feedback linearisation as shown in [3]. However, the complex friction forces acting between the snake-like robot and the ground are difficult to model accurately, yet, these friction forces play a crucial role in achieving undulatory locomotion. More precisely, assume only nominal values $\tilde{c}_n$, $\tilde{c}_t$ and $\tilde{c}_p$ of the friction coefficients c_n, c_t and c_p are given and consider the input law

$$\upsilon = \upsilon_{fl} = m(DD^T)^{-1} \left(\bar{u} + \frac{\tilde{c}_n}{m} v_\phi - \frac{\tilde{c}_p}{m} v_t A D^T \phi \right). \tag{4.80}$$

If $\tilde{c}_\sigma = c_\sigma$, for $\sigma \in \{n, t, p\}$, then (4.78e) with v as given in (4.80) reduces to

$$\dot{v}_\phi = \bar{u},$$

and $\bar{u}$ can be designed to ensure ϕ_i tracks $\phi_{i,\mathrm{loc}}$, for $i = 1, \ldots, N_l - 1$. In [212, Section V] it is shown that the choice

$$\bar{u} = \ddot{\phi}_{\mathrm{loc}} + k_p(\phi_{\mathrm{loc}} - \phi) + k_d(\dot{\phi}_{\mathrm{loc}} - v_\phi), \tag{4.81}$$

where $\phi_{\mathrm{loc}} = [\phi_{1,\mathrm{loc}}, \ldots, \phi_{N_l-1,\mathrm{loc}}]^\top$, with $k_p > 0$ and $k_d > 0$, ensures that $\lim_{t\to\infty}(\phi_{\mathrm{loc}}(t) - \phi(t)) = 0$, for any initial condition $\xi(0)$. Consider instead the case in which $\tilde{c}_\sigma \neq c_\sigma$, for $\sigma \in \{n, t, p\}$. The terms including the friction coefficients in (4.78e) no longer cancel out. Focusing on the subsystem described by the actuated states ϕ and v_ϕ, namely (4.78a), (4.78e), the dynamics of the "actuated subsystem" with v as given in (4.80) are described by

$$\begin{aligned}\dot{\phi} &= v_\phi\,,\\ \dot{v}_\phi &= \frac{\tilde{c}_n - c_n}{m} v_\phi + \frac{c_p - \tilde{c}_p}{m} v_t A D^T \phi + \bar{u}\,,\end{aligned} \tag{4.82}$$

Linearising (4.82) around the reference trajectories

$$\begin{aligned}\phi(t) = \phi_{\mathrm{loc}}(t) &= \begin{bmatrix}\phi_{1,\mathrm{loc}}(t) & \ldots & \phi_{N_l-1,\mathrm{loc}}(t)\end{bmatrix}^\top,\\ v_\phi(t) &= \dot{\phi}_{\mathrm{loc}}(t),\\ \bar{u}(t) &= \ddot{\phi}_{\mathrm{loc}}(t),\end{aligned}$$

and discretising via zero-order hold with time step Δ results in an LTV system of the form (4.1a) with unknown time-varying matrices $A(k)$ and $B(k)$. In this case, the result of Theorem 4.1.3 provides a method to design a control law similar to (4.81), using data in place of exact knowledge of the friction coefficients, which ensures that the joint angle trajectories $\phi(t)$ stay within a bound of the reference $\phi_{\mathrm{loc}}(t)$. Namely, let

$$\bar{u}(t) = \ddot{\phi}_{\mathrm{loc}}(t) + u(t), \tag{4.83}$$

where $u(t)$ is obtained via zero-order hold from $u(k) = K(k)x(k)$, with

$$x(k) = \begin{bmatrix}\phi(k\Delta) - \phi_{\mathrm{loc}}(k\Delta) & v_\phi(k\Delta) - \dot{\phi}_{\mathrm{loc}}(k\Delta)\end{bmatrix}^\top,$$

and where $K(k)$ is designed using Theorem 4.1.3. In [32] simulation studies are conducted comparing the performance of a controller of the form (4.83) designed using data to a controller of the form (4.81) with the gains k_p, k_d chosen as in [3, Section 6.10]. For an average error of 30% between the nominal friction coefficients used in (4.80) and the actual friction coefficients it is shown that the controller choice of (4.83) for $\bar{u}$ outperforms the choice (4.81) without re-tuning the gains. While strategic tuning methods exist (see e.g. [96]) and can be used to improve the performance of (4.81) for the case $\tilde{c}_\sigma \neq c_\sigma$, for $\sigma \in \{n, t, p\}$, the proposed data-driven methods can have benefits from a user point of view by providing a systematic control design approach in the presence of inaccurate knowledge of the friction coefficients. Instead of repeated trials involved in the tuning process, the data-driven approach based on Theorem 4.1.3 only requires a predefined number of experiments for data collection. Another benefit of the resulting time-varying feedback controller in (4.83) compared to the static feedback law in (4.81) is that it can capture any time-variation in the dynamics. For example, consider a control task which involves moving the centre of gravity of the snake-like robot from an initial position on one type of surface to a final position on a different type of surface with different friction properties. The static gains k_p and k_d in (4.81) need to be tuned for a each type of surface and if a single controller is to be used for the control task, then this represents a compromise. In contrast to this, the changes in friction coefficients c_n, c_t and c_p along the robot's trajectory are repeatable time-variations, which are captured by the ensemble of data collection experiments and are hence incorporated into the control design via the data-driven approach based on Theorem 4.1.3.

4.2 Data-driven cost representation

While the previous section focuses on control design in the context of unknown system dynamics, this section instead considers the problem of designing optimal controllers if the objective functions are not known a priori. Inverse optimal control [164, 81] addresses this problem by reconstructing cost functions from expert data. In this section the problem is considered from a different point of view. Namely, the cost of an LQR problem is reconstructed using a finite, non-optimal, open-loop data sequence of the system state, the input and a performance variable. Rather than identifying the cost weights from the data and then solving the classical LQR problem, this section builds upon the results presented in [95] and it is shown that a similar approach can be employed to represent the quadratic cost function directly using data. In combination with the results from [95], the presented results allow to solve LQR problems involving both unknown dynamics *and* unknown (quadratic) cost via a purely data-dependent convex optimisation problem.

The remainder of the section is organised as follows. The considered problem is defined in Section 4.2.1. The data-driven representation of cost functions in the context of LQR problems is introduced in Section 4.2.2. This representation is used in Section 4.2.3 to formulate and solve the LQR problem via an SDP with purely data-dependent constraints.

4.2.1 Problem formulation

Consider the LTI system (2.17), namely

$$x(k+1) = Ax(k) + Bu(k),$$

with $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$, and the problem of determining a stabilising state-feedback control law for the linear dynamical system, which minimises an *unknown* quadratic cost functional. Instead of requiring knowledge of an expert strategy (as is typically required in inverse optimal control), assume it is possible to perform open-loop experiments on the system and collect “non-expert” data of the input $u(k)$, state $x(k)$ and a performance output over the finite time interval $k = 0, \ldots, T$, $T \in \mathbb{N}$. The performance output is defined as

$$z(k) = Cx(k) + Du(k), \tag{4.84}$$

where $z \in \mathbb{R}^q$ and C, D are *unknown* constant matrices such that $D^\top D \succ 0$. Let $\mathcal{A} = A - B(D^\top D)^{-1}D^\top C$ and $\mathcal{C} = C^\top C - C^\top D(D^\top D)^{-1}D^\top C$ and consider the following assumption.

Assumption 4.2.1. The (unknown) matrices A, B, C and D are such that the pair (A, B) is stabilisable and the pair $(\mathcal{A}, \mathcal{C})$ is observable[10].

The considered problem is formalised in the following statement.

Problem 4.2.1. Consider the system (2.17) and the cost functional

$$J\left(x(0), u(\cdot)\right) = \sum_{k=0}^{\infty} z(k)^\top z(k), \tag{4.85}$$

with $z(k)$ as defined in (4.84). Suppose the system matrices A, B *and* cost matrices C, D are unknown, but that finite data sequences of the state $x_{d,[0,T]}$, input $u_{d,[0,T-1]}$ and performance output $z_{d,[0,T-1]}$ are available. Determine the control input $u^\star(x)$, which renders the zero equilibrium of the system (2.17) with $u = u^\star$ asymptotically stable and which is such that $J(x(0), u^*) \leq J(x(0), u)$, for all u.

While the focus lies on the data-driven representation of quadratic cost functions, the available data is sufficient to also represent the system dynamics directly using data (as in [95]). To consider the most general case, the system dynamics are hence treated as unknown[11].

4.2.2 Cost representation

Rather than using the collected data to identify the unknown matrices, it is proposed to parameterise the cost function with data – similarly to how the system dynamics are parameterised in [95] – in order to solve the problem directly using data. Consider the data matrices U_-, X_- and X_+ as defined in (2.18) with $k_0 = 0$, namely,

$$\begin{aligned} U_- &= \begin{bmatrix} u_d(0) & \dots & u_d(T-1) \end{bmatrix}, \\ X_- &= \begin{bmatrix} x_d(0) & \dots & x_d(T-1) \end{bmatrix}, \\ X_+ &= \begin{bmatrix} x_d(1) & \dots & x_d(T) \end{bmatrix}, \end{aligned}$$

as well as

$$Z_- = \begin{bmatrix} z_d(0) & \dots & z_d(T-1) \end{bmatrix}. \tag{4.86}$$

Corollary 4.2.1. *Suppose the rank condition* (2.19) *holds, i.e.*

$$\mathit{rank}\left(\begin{bmatrix} X_- \\ U_- \end{bmatrix}\right) = n + m,$$

then the performance output (4.84) *in closed-loop with* $u(k) = K(k)x(k)$ *can equivalently be represented*

[10] Assumption 4.2.1 ensures that the algebraic Riccati equation associated with the optimal control problem (Problem 4.2.1) has a unique, stabilising solution, see e.g [52].

[11] It is worth noting that the results are applicable and relevant even if the system matrices A and B are known.

as

$$z(k) = Z_- G x(k), \tag{4.87}$$

where G satisfies (2.21)*, namely*

$$\begin{bmatrix} I_n \\ K \end{bmatrix} = \begin{bmatrix} X_- \\ U_- \end{bmatrix} G.$$

Proof. Consider the data matrices U_-, X_- and Z_- and note that

$$Z_- = CX_- + DU_-.$$

Since (2.19) holds,

$$C + DK = \begin{bmatrix} C & D \end{bmatrix} \begin{bmatrix} I_n \\ K \end{bmatrix} = Z_- G, \tag{4.88}$$

with G satisfying (2.21), analogous to Theorem 2.2.1. □

4.2.3 Optimal Control

In the following, the data-driven performance output representation introduced in Corollary 4.2.1 is combined with the system representation in Theorem 2.2.1 to formulate and solve LQR problems in the context of unknown cost criteria and unknown system dynamics via a data-dependent SDP.

Theorem 4.2.1. *Consider Problem 4.2.1. Suppose Assumption 4.2.1 holds and that the available data is such that the condition* (2.19) *is satisfied for the matrices* (2.18)*. Then, the optimal control input is given by $u^\star = K^\star x$, with $K^\star$ given by*

$$K^\star = U_- H^\star \left(X_- H^\star\right)^{-1}, \tag{4.89}$$

with $H^\star$ the solution of

$$\begin{aligned} \min_{\gamma, S, H, O} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{Tr}\left(O\right) \leq \gamma, \\ & \begin{bmatrix} S - I_n & X_+ H \\ H^\top X_+^\top & S \end{bmatrix} \succeq 0, \\ & \begin{bmatrix} O & Z_- H \\ H^\top Z_-^\top & S \end{bmatrix} \succeq 0, \\ & X_- H = S. \end{aligned} \tag{4.90}$$

Proof. Given Assumption 4.2.1, the solution of the LQR problem defined by the cost function (4.85), (4.84) and the dynamics (2.17) is unique and given by a static state feedback law of the form $u^\star(k) = K^\star x(k)$, with

$$K^\star = -\left(D^\top D + B^\top P B\right)^{-1}\left(D^\top C + B^\top P A\right),$$

and $P = P^\top \succ 0$ satisfying the algebraic Riccati equation

$$P = (C + DK^\star)^\top (C + DK^\star) + (A + BK^\star)^\top P (A + BK^\star),$$

see e.g. [52]. From Lagrange duality [213], the optimal feedback gain $K^\star$ can equivalently be found via the optimisation

$$\begin{aligned} \min_{K,S} \quad & \mathbf{Tr}\left((C + DK)\, S\, (C + DK)^\top\right) \\ \text{s.t.} \quad & S - I_n - (A + BK)\, S\, (A + BK)^\top \succeq 0, \end{aligned} \tag{4.91}$$

where $S = S^\top \succeq I_n$ is the controllability Gramian [214]. The SDP (4.90) is obtained by introducing the results of Theorem 2.2.1 and Corollary 4.2.1, i.e. by introducing $(A+BK) = X_+G$ and $(C+BK) = Z_-G$, and by converting the problem into a convex programme via the the change of variable $H := GS$ and using the Schur complement. Note that the equality constraint $S = X_-H$ comes from the upper row block in (2.21). The data representation of the optimal gain (4.89) is given via the lower row block in (2.21). □

Remark 4.2.1. In the continuous-time case, the closed-loop system can be represented as[12] $\dot{x}(t) = X_+Gx(t)$, and the control law as $u(t) = U_-Gx(t)$, with G satisfying (2.21) with

$$\begin{aligned} X_- &= \begin{bmatrix} x_d(0) & \dots & x_d((T-1)\Delta) \end{bmatrix}, \\ U_- &= \begin{bmatrix} u_d(0) & \dots & u_d((T-1)\Delta) \end{bmatrix}, \\ X_+ &= \begin{bmatrix} \dot{x}_d(0) & \dots & \dot{x}_d((T-1)\Delta) \end{bmatrix}, \end{aligned}$$

where Δ is the sampling time [95]. Similarly, (4.88) holds with

$$Z_- = \begin{bmatrix} z_d(0) & \dots & z_d((T-1)\Delta) \end{bmatrix}.$$

Hence, the optimal control strategy can be determined following analogous steps as in Theorem 4.2.1, considering the continuous-time equivalent of (4.91). Despite being designed using sampled data, the resulting controller is optimal for the continuous-time system, not the equivalent sampled-data model.

[12]Note that this assumes that the derivative of the state is available for measurement, which is not usually the case and filtering or numerical methods may be required to approximate the derivative.

Remark 4.2.2. While the focus in this section is on LTI systems, the data-driven cost representation can be extended to the time-varying case considered in Section 4.1 if an ensemble of performance output data is available. Namely, consider the matrices (4.7a), (4.7b) and

$$\bar{Z}(k) = \begin{bmatrix} z_{d,1}(k) & \dots & z_{d,L}(k) \end{bmatrix},$$

for $k = 0, \dots, T-1$. If the rank condition (4.13) holds, then the closed-loop performance output can equivalently be represented as

$$z(k) = \bar{Z}(k)G(k)x(k),$$

with $G(k)$ satisfying (4.15), for $k = 0, \dots, T-1$.

4.3 Conclusion

A data-driven representation of closed-loop LTV systems under state-feedback is employed to design feedback controllers ensuring that the resulting closed-loop trajectories satisfy certain boundedness, performance and robustness criteria via the formulation of convex feasibility or optimisation problems involving data-dependent LMIs. Both the noise-free case and the case in which the data and the system are affected by process and measurement noise are considered. Approaches to tackle challenges arising for large time horizons are proposed and special insights are also provided for the case of periodically time-varying systems. The results are illustrated and motivated via both numerical examples and practical examples involving a power converter and locomotion of a snake-like robot.

For LTI systems, an approach to represent unknown cost functionals in the context of LQR problems directly using finite, open-loop, “non-expert” data is proposed. In combination with the data-based representation of unknown LTI systems presented in [95], this makes it possible to solve LQR problems with both unknown dynamics *and* unknown cost via a purely data-dependent SDP. While this is an interesting result by itself, it is also relevant to a specific class of LQ dynamic games with asymmetric information structure. This is addressed and illustrated via an example involving human-robot interaction in Chapter 5.

Chapter 5

Data-driven methods for dynamic games

In Chapter 3, infinite-horizon dynamic games and in particular their feedback Nash equilibrium solutions, are studied under the assumption that each player has complete knowledge of the dynamics of the entire system and the performance criteria of all players. While this constitutes the "classical" game formulation, dynamic game problems arise in a variety of applications, such as multi-agent systems [29], cyber-physical systems [77] or human-robot systems [20], in which it is possible or even likely that each player has access to different and typically partial information regarding the system dynamics and performance criteria of the other players. In this chapter, this challenge is addressed by considering games with different incomplete information structures and introducing data-driven methods from Chapter 2 and Chapter 4 in the context of results of Chapter 3 to overcome incomplete information. Firstly, in Section 5.1 it is shown that the results of Section 4.2 are relevant to a specific class of LQ dynamic games with asymmetric information structure, in which one of the players does not know the performance objectives of the other players, and may not know the system dynamics. Secondly, in Section 5.2 the focus lies on games with incomplete information in the sense that each player lacks knowledge of the performance objectives of the other players and the system dynamics. It is shown that using data in the context of the iterative methods proposed in Section 3.4 the players can jointly converge to a feedback Nash equilibrium.

5.1 A class of LQ games with partially unknown information

From distributed control [29], human-robot interaction [20] to cyber-security [77], there are various settings in which each player may have access to different and potentially incomplete information in the context of dynamic game problems. In this section, a class of deterministic, non-cooperative, nonzero-sum, infinite-horizon, LQ, discrete-time dynamic games is considered in which all players but one have access to full system and cost information, whereas the remaining player only has knowledge of their own control objective, but lacks knowledge regarding the objectives of the other players and may not know the system dynamics. The objective is to determine a feedback Nash equilibrium solution of the game. While the players with complete information are faced with a classical game problem, the remaining player lacks the required information to determine a Nash equilibrium. If the "fully informed" players adhere to strategies corresponding to a feedback Nash equilibrium, the problem of determining the corresponding equilibrium strategy for the "uninformed" player is reduced to an optimal control problem, with unknown terms appearing in *both* the system dynamics and in the cost function. It is demonstrated that, given appropriate measurements, the player can determine the Nash equilibrium strategy based solely on collected data utilising the results in Section 4.2, which combine data-driven results introduced in [95] and recalled in Chapter 2 with a data-driven cost representation.

To demonstrate the relevance of the presented results, a practically motivated example involving simulations of a *human-robot system* is considered. In [20] human-robot interactions are modelled as a two-player LQ game, inspired by evidence that human behaviour in such interactive settings can be modelled as Nash equilibrium strategies of a game [215]. However, the contact robot cannot know the performance criteria of the human *a priori* and may need to react appropriately to different human operators (with different unknown dynamic behaviours). Hence, the game cannot be solved using classical methods and it is demonstrated that the presented results provide a strategy to overcome this difficulty.

The remainder of this section is structured as follows. In Section 5.1.1 the considered problem is specified, before a data-driven solution approach is proposed in Section 5.1.2. The results are illustrated via simulations involving human-robot interactions in Section 5.1.3.

5.1.1 Problem formulation

Consider the system (3.1), namely,

$$x(k+1) = Ax(k) + \sum_{i=1}^{N} B_i u_i(k),$$

with $x \in \mathbb{R}^n$, which is influenced by the control actions of N players. Let each player i be associated with a performance output

$$z_i(k) = C_i x(k) + \sum_{j=1}^{N} D_{ij} u_j(k), \tag{5.1}$$

where C_i and D_{ij} are constant matrices of appropriate dimension, such that $D_{ii}^\top D_{ii} \succ 0$, for $i = 1, \ldots, N$, $j = 1, \ldots, N$. Assume each player i seeks to minimise the cost functional

$$J_i\left(x(0), u_1(\cdot), \ldots, u_N(\cdot)\right) = \sum_{k=0}^{\infty} z_i(k)^\top z_i(k), \tag{5.2}$$

via the choice of control action $u_i \in \mathbb{R}^{m_i}$, for $i = 1, \ldots, N$.

Remark 5.1.1. Note that the cost functionals (5.2), for $i = 1, \ldots, N$, are more general than the cost functionals typically encountered in the dynamic games literature (see e.g. [9]), more precisely, cost functionals of the form (3.2), $i = 1, \ldots, N$, as considered in Chapter 3. In the special case in which $C_i^\top D_{ij} = 0$ and $D_{ij}^\top D_{il} = 0$, for $i = 1, \ldots, N$, $j = 1, \ldots, N$, $l = 1, \ldots, N$, $l \neq j$, the described problem corresponds to the game (3.1), (3.2), $i = 1, \ldots, N$, considered in Chapter 3. In addition to being in line with related literature, this form of cost functionals has been chosen in Chapter 3 to streamline the notation. In this section, the more general case is considered instead. This is not only motivated by the application example considered in Section 5.1.3, but also more generally by the fact that discretised game problems, which are expected to be of particular relevance in the context of determining equilibrium strategies directly using sampled data, are typically of this form. More precisely, consider a continuous-time LQ dynamic game with the cost functionals being the continuous-time version of (3.2), $i = 1, \ldots, N$, i.e. there are no cost terms penalising weighted inner products between the state and a player's input or between different players' inputs. If the problem is discretised using zero-order hold, then the equivalent discrete-time cost is typically of the form (5.2), for $i = 1, \ldots, N$, i.e. the off-diagonal terms $C_i^\top D_{ij}$ and $D_{ij}^\top D_{il}$, for $i = 1, \ldots, N$, $j = 1, \ldots, N$, $l = 1, \ldots, N$, $l \neq j$, are in general nonzero, see e.g. [216].

With the aim of determining admissible strategies in the sense of Definition 3.3.1 for the control actions u_i, $i = 1, \ldots, N$, which constitute a feedback Nash equilibrium (see Definition 2.1.2) solution of the game (3.1), (5.2), $i = 1, \ldots, N$, the result of Theorem 3.1.1 is revisited and revised for the more general cost (5.2), $i = 1, \ldots, N$, in the following result.

Corollary 5.1.1. *Consider the game* (3.1), (5.2), $i = 1, \ldots, N$*. The set of strategies*

$$\left\{\phi_1^\star(x(k)), \ldots, \phi_N^\star(x(k))\right\},$$

where $\phi_i^\star(x(k)) = K_i^\star x(k)$*, for* $i = 1, \ldots, N$*, constitutes a feedback Nash equilibrium solution of the game if and only if* (3.4) *holds, i.e.*

$$\rho\left(A + \sum_{j=1}^{N} B_j K_j^\star\right) < 1,$$

and there exist $P_i^\star = P_i^{\star\top} \succeq 0 \in \mathbb{R}^{n\times n}$ *satisfying*

$$P_i^\star = \left(C_i + \sum_{j=1}^{N} D_{ij}K_j^\star\right)^\top \left(C_i + \sum_{j=1}^{N} D_{ij}K_j^\star\right) + \left(A + \sum_{j=1}^{N} B_jK_j^\star\right)^\top P_i^\star \left(A + \sum_{j=1}^{N} B_jK_j^\star\right), \quad (5.3)$$

for $i = 1, \dots, N$, *and*

$$\begin{bmatrix} D_{11}^\top D_{11} + B_1^\top P_1^\star B_1 & \dots & D_{11}^\top D_{1N} + B_1^\top P_1^\star B_N \\ \vdots & \ddots & \vdots \\ D_{NN}^\top D_{N1} + B_N^\top P_N^\star B_1 & \dots & D_{NN}^\top D_{NN} + B_N^\top P_N^\star B_N \end{bmatrix} \begin{bmatrix} K_1^\star \\ \vdots \\ K_N^\star \end{bmatrix} = - \begin{bmatrix} D_{11}^\top C_1 + B_1^\top P_1^\star A \\ \vdots \\ D_{NN}^\top C_N + B_N^\top P_N^\star A \end{bmatrix}. \quad (5.4)$$

The feedback Nash equilibrium is such that the equilibrium cost incurred by player i starting from initial condition $x(0)$ *is* $J_i^\star = J_i(x(0), \phi_1^\star(x(k)), \dots, \phi_N^\star(x(k))) = x(0)^\top P_i^\star x(0)$.

Proof. The proof is analogous to that of Theorem 3.1.1, considering the cost functionals (5.2) instead of (3.2), for $i = 1, \dots, N$. More precisely, let $\{\bar{K}_1^\star, \dots, \bar{K}_N^\star\}$ and $\{\bar{P}_1^\star, \dots, \bar{P}_N^\star\}$ satisfy (3.4), (5.3), $i = 1, \dots, N$, and (5.4). Assuming that the actions of players j, $j = 2, \dots, N$, are fixed at $u_j(k) = \bar{K}_j^\star x(k)$, consider the minimisation of the cost function (5.2), $i = 1$, subject to the system dynamics (3.1) by player 1, namely the minimisation of

$$\begin{aligned} J_1(x(0), u_1(\cdot), \bar{K}_2^\star x(k), \dots, \bar{K}_N^\star x(k)) &= \sum_{k=0}^{\infty} \left(\underbrace{\left(C_1 + \sum_{j=2}^{N} D_{1j}\bar{K}_j^\star\right)}_{\bar{C}_1} x(k) + D_{11}u_1(k) \right)^\top \\ &\qquad \times \left(\left(C_1 + \sum_{j=2}^{N} D_{1j}\bar{K}_j^\star\right) x(k) + D_{11}u_1(k) \right) \\ &= \sum_{k=0}^{\infty} \left[x(k)^\top u_1(k)^\top\right] \begin{bmatrix} \bar{C}_1^\top \bar{C}_1 & \bar{C}_1^\top D_{11} \\ D_{11}^\top \bar{C}_1 & D_{11}^\top D_{11} \end{bmatrix} \begin{bmatrix} x(k) \\ u(k) \end{bmatrix}, \end{aligned} \quad (5.5)$$

subject to (3.8), namely,

$$x(k+1) = \underbrace{\left(A + \sum_{j=2}^{N} B_j\bar{K}_j^\star\right)}_{\bar{A}_1} x(k) + B_1u_1(k).$$

This constitutes an LQR problem (with weighting of state-input inner product) for player 1. Note that by assumption, there exists a stabilising solution $P^{OC} = \bar{P}_1^\star$ satisfying the algebraic Riccati equation

$$P^{OC} = \bar{C}_1^\top \bar{C}_1 + \bar{A}_1^\top P^{OC} \bar{A}_1 - \left(\bar{C}_1^\top D_{11} + \bar{A}_1^\top P^{OC} B_1\right) \left(D_{11}^\top D_{11} + B_1^\top P^{OC} B_1\right)^{-1} \left(D_{11}^\top \bar{C}_1 + B_1^\top P^{OC} \bar{A}_1\right), \tag{5.6}$$

associated with this LQR problem. This follows by rearranging the first line of (5.4) and inserting it into (5.3), $i = 1$. With the strategies of players j, $j = 2, \dots, N$, fixed, the optimal strategy for player 1 is hence

$$u_1^{OC}(k) = -\left(D_{11}^\top D_{11} + B_1^\top P^{OC} B_1\right)^{-1} \left(D_{11}^\top \bar{C}_1 + B_1^\top P^{OC} \bar{A}_1\right) x(k) = \bar{K}_1^\star x(k), \tag{5.7}$$

and the optimal cost is $J_1(x(0), \bar{K}_1^\star x(k), \dots, \bar{K}_N^\star x(k)) = x(0)^\top P^{OC} x(0) = x(0)^\top \bar{P}_1^\star x(0)$, see e.g. [52]. Hence, (3.11) holds, i.e.

$$J_1(x(0), \bar{K}_1^\star x(k), \dots, \bar{K}_N^\star x(k)) \leq J_1(x(0), u_1(\cdot), \bar{K}_2^\star x(k), \dots, \bar{K}_N^\star x(k)),$$

for all admissible $\left\{u_1(\cdot), \bar{K}_2^\star x(k), \dots, \bar{K}_N^\star x(k)\right\}$. Conversely, if the set of gains $\left\{\bar{K}_1^\star, \dots, \bar{K}_N^\star\right\}$ corresponds to a feedback Nash equilibrium solution of the game (3.1), (5.2), $i = 1, \dots, N$, then by Definition 2.1.2, (3.11) holds for all admissible $\left\{u_1(\cdot), \bar{K}_2^\star x(k), \dots, \bar{K}_N^\star x(k)\right\}$, and with the actions of players j, $j = 2, \dots, N$, fixed at $u_j(k) = \bar{K}_j^\star x(k)$, (5.7) with P^{OC} the solution of (5.6) is the unique stabilising optimal control action for player 1 solving the LQR problem (3.8), (5.5), see e.g. [52]. This implies that there exists $\bar{P}_1^\star = P^{OC}$ such that (5.3), $i = 1$, and the first line of (5.4) hold. The proof is concluded via analogous arguments for players j, $j = 2, \dots, N$. □

In Corollary 5.1.1 feedback Nash equilibrium solutions of the game (3.1), (5.2), $i = 1, \dots, N$, are characterised via the stabilising solutions of a set of coupled algebraic equations, namely (5.3), $i = 1, \dots, N$, and (5.4), which may also admit solutions which do not render the closed-loop system stable. In Corollary 3.1.1 in Section 3.1, sufficient conditions depending only on the system and cost matrices are provided for the solutions of the equivalent coupled equations (3.5), $i = 1, \dots, N$, and (3.6), corresponding to the game (3.1), (3.2), $i = 1, \dots, N$, to be stabilising. Due to the additional inner product terms between the state and the inputs as well as between the inputs of different players in the more general cost functional (5.2), equivalent conditions for the game (3.1), (5.2), $i = 1, \dots, N$, depend on the solution $\{K_1^\star, \dots, K_N^\star\}$. For example, analogous to the conditions provided in [9, Proposition 6.3] (for the game (3.1), (3.2), $i = 1, \dots, N$), a solution $\{K_1^\star, \dots, K_N^\star\}$ and $\{P_1^\star, \dots, P_N^\star\}$ to (5.3), $i = 1, \dots, N$, and (5.4) is such that (3.4) holds and the corresponding feedback strategies $\phi_i^\star(x(k)) = K_i^\star x(k)$, $i = 1, \dots, N$, constitute a feedback Nash equilibrium of the game (3.1), (5.2), $i = 1, \dots, N$, if the pair

$$\left(\bar{A}_i, B_i\right),$$

where $\bar{A}_i = A + \sum_{j=1,j\neq i}^{N} B_j K_j^\star$, is stabilisable and the pair

$$\left(\left(\bar{A}_i - B_i\left(D_{ii}^\top D_{ii}\right)^{-1} D_{ii}^\top \bar{C}_i\right),\ \left(\bar{C}_i^\top \bar{C}_i - \bar{C}_i^\top D_{ii}\left(D_{ii}^\top D_{ii}\right)^{-1} D_{ii}^\top \bar{C}_i\right)\right),$$

where $\bar{C}_i = C_i + \sum_{j=1,j\neq i}^{N} D_{ij} K_j^\star$, is detectable, for $i = 1,\ldots,N$. In the following result, alternative sufficient conditions are provided. While these may be less general, they are more easily verified given a solution $\{K_1^\star,\ldots,K_N^\star\}$ and $\{P_1^\star,\ldots,P_N^\star\}$ to (5.3), $i = 1,\ldots,N$, and (5.4).

Corollary 5.1.2. *Let* $\{K_1^\star,\ldots,K_N^\star\}$ *and* $\{P_1^\star,\ldots,P_N^\star\}$, *where* $P_i^\star = P_i^{\star\top} \succeq 0$, *for* $i = 1,\ldots,N$, *be a solution to* (5.3), $i = 1,\ldots,N$, *and* (5.4). *If*

$$\sum_{i=1}^{N} P_i^\star \succ 0, \tag{5.8a}$$

and

$$\sum_{i=1}^{N}\left(C_i + \sum_{j=1}^{N} D_{ij} K_j^\star\right)^\top \left(C_i + \sum_{j=1}^{N} D_{ij} K_j^\star\right) \succ 0, \tag{5.8b}$$

then $\{K_1^\star,\ldots,K_N^\star\}$ *is such that* (3.4) *holds and the corresponding strategies* (3.3), *i.e.* $\phi_i^\star(x(k)) = K_i^\star x(k)$, $i = 1,\ldots,N$, *constitute a feedback Nash equilibrium solution of the game* (3.1), (5.2), $i = 1,\ldots,N$.

Proof. Consider the sum over i of (5.3), for $i = 1,\ldots,N$, namely

$$\begin{aligned}\sum_{i=1}^{N} P_i^\star = \sum_{i=1}^{N}\left(C_i + \sum_{j=1}^{N} D_{ij} K_j^\star\right)^\top &\left(C_i + \sum_{j=1}^{N} D_{ij} K_j^\star\right) \\ &+ \left(A + \sum_{j=1}^{N} B_j K_j^\star\right)^\top \left(\sum_{i=1}^{N} P_i^\star\right)\left(A + \sum_{j=1}^{N} B_j K_j^\star\right),\end{aligned} \tag{5.9}$$

and the candidate Lyapunov function $W(x(k)) = x(k)^\top \left(\sum_{i=1}^{N} P_i^\star\right) x(k)$. If the conditions (5.8) hold, then (5.9) implies that $W(x(k+1)) - W(x(k)) < 0$, for all $x \neq 0$, and hence that (3.4) holds. □

To characterise feedback Nash equilibrium solutions of the game (3.1), (5.2), $i = 1,\ldots,N$, in terms of the coupled matrix equations (5.3), $i = 1,\ldots,N$, and (5.4) – and to obtain their solutions and thereby the Nash equilibrium strategies – it is required that each player i has full knowledge of the system dynamics (3.1), the objective function J_i given in (5.2) *and* the objective functions J_j, $j = 1,\ldots,N$, $j \neq i$, of all the "opponents". Consider instead the case in which different information is available to different players. More precisely, let one player (player N) only have knowledge regarding the own objective J_N, but have no knowledge of the objectives J_j, $j = 1,\ldots,N-1$, of the remaining players and may not know the system dynamics (3.1). All other players j, $j = 1,\ldots,N-1$, instead have

Table 5.1: Information available to each player. The parentheses indicate information which may be available but is not required for the proposed solution as detailed in Section 5.1.2.

	Player 1	Player 2
Dynamics:	A, B_1, B_2	(A, B_1, B_2)
Cost:	C_1, D_{11}, D_{12}, C_2, D_{21}, D_{22}	(C_2, D_{21}, D_{22})

full knowledge of the objectives of all players (including themselves), as well as the system dynamics. For ease of exposition, the results in this section are presented for the two-player case (i.e. $N = 2$), however, they trivially extend to the N-player case, provided only one of the players is faced with limited information. Under the given information structure, which is summarised in Table 5.1, player 1 is faced with a classical dynamic game (3.1), (5.2), $i = 1, 2$, and can determine a feedback gain $K_1^\star$ corresponding to a Nash equilibrium solution by solving (3.4), (5.3), $i = 1, 2$, and (5.4)[1]. Player 2, on the other hand, lacks part of the required information.

Assumption 5.1.1. Let $\{K_1^\star, K_2^\star\}$ and $\{P_1^\star, P_2^\star\}$, where $P_i^\star = P_i^{\star\top} \succeq 0$, for $i = 1, 2$, be a solution to (3.4), (5.3), $i = 1, 2$, and (5.4), such that $P_2^\star \succ 0$. Assume player 1 adheres to the corresponding feedback Nash equilibrium strategy, i.e. $u_1(k) = \phi_1^\star(x(k)) = K_1^\star x(k)$, irrespective of the actions of player 2.

This results in the following problem.

Problem 5.1.1. Consider the game (3.1), (5.2), $i = 1, 2$, and let Assumption 5.1.1 hold. Given the underlying information structure as specified in Table 5.1, determine the corresponding feedback Nash equilibrium strategy $\phi_2^\star(k) = K_2^\star x(k)$ of player 2 .

5.1.2 Data-driven Nash equilibrium solution

In the following, Problem 5.1.1 is addressed by demonstrating how player 2 can utilise available data to compensate for lack of cost and system information. To this end, note that under the given Assumptions, the problem faced by player 2 can be reduced to an LQR problem with unknown terms appearing in both the system dynamics and the cost function, as detailed in the following result.

Proposition 5.1.1. *Consider Problem 5.1.1. The problem of determining the Nash Equilibrium strategy $\phi_2^\star(x(k))$ of player 2 constitutes an LQR problem with unknown terms in both the cost functional and in the dynamics and $\phi_2^\star(x(k))$ is the unique solution to this LQR problem.*

Proof. With the action of player 1 fixed at $u_1(k) = K_1^\star x(k)$, the cost functional (5.2), $i = 2$, which

[1]Note that solving the coupled algebraic matrix equations characterising Nash equilibrium solutions of nonzero-sum multi-player games is generally difficult [35], as also discussed in Chapter 3. However, it is assumed here that an exact solution can be determined by player 1.

player 2 aims to minimise via the action u_2 becomes

$$J_2\left(x(0), K_1^\star x(k), u_2(\cdot)\right) = \sum_{k=0}^{\infty} \left(\bar{C}_2 x(k) + D_{22} u_2(k)\right) \left(\bar{C}_2 x(k) + D_{22} u_2(k)\right), \tag{5.10}$$

and the dynamics (3.1) are perceived by player 2 as

$$x(k+1) = \bar{A}_2 x(k) + B_2 u(k), \tag{5.11}$$

where $\bar{A}_2 = A + B_1 K_1^\star$ and $\bar{C}_2 = C_2 + D_{21} K_1^\star$, as defined in Section 5.1.1. This constitutes an LQR problem (with weighting of state-input inner product). There exists a unique solution $P^{OC} = P^{OC\top} \succ 0$ to the algebraic Riccati equation

$$P^{OC} = \bar{C}_2^\top \bar{C}_2 + \bar{A}_2^\top P^{OC} \bar{A}_2 - \left(\bar{C}_2^\top D_{22} + \bar{A}_2^\top P^{OC} B_2\right) \left(D_{22}^\top D_{22} + B_2^\top P^{OC} B_2\right)^{-1} \left(D_{22}^\top \bar{C}_2 + B_2^\top P^{OC} \bar{A}_2\right), \tag{5.12}$$

associated with this LQR problem and the optimal strategy is given by

$$u_2^{OC}(k) = -\left(D_{22}^\top D_{22} + B_2^\top P^{OC} B_2\right)^{-1} \left(D_{22}^\top \bar{C}_2 + B_2^\top P^{OC} \bar{A}_2\right) x(k) = K^{OC} x(k), \tag{5.13}$$

if and only if the pair $\left(\bar{A}_2 - B_2 (D_{22}^\top D_{22})^{-1} D_{22}^\top \bar{C}_2,\ \bar{C}_2^\top \bar{C}_2 - \bar{C}_2^\top D_{22} (D_{22}^\top D_{22})^{-1} D_{22}^\top \bar{C}_2\right)$ is observable and the pair $(\bar{A}_2, B_2)$ is stabilisable (see e.g. [52]). Recall from Assumption 5.1.1 that $K_1^\star$ is such that there exist $K_2^\star$, $P_2^\star = P_2^{\star\top} \succ 0$ satisfying (3.4), (5.3), for $i = 2$, and (5.4) with $N = 2$, and note that (5.3), $i = 2$, and the second row of (5.4) give (5.12), (5.13) with $P^{OC} = P_2^\star$ and $K^{OC} = K_2^\star$. Hence, the conditions for a unique solution are satisfied and $\phi_2^\star(x(k)) = K_2^\star x(k)$ is the unique optimal strategy. Since player 2 lacks information regarding the control objective of player 1, $K_1^\star$ and consequently the matrices $\bar{A}_2$ and $\bar{C}_2$ are unknown to player 2. □

With this reformulation in place, it is proposed to use the results of Section 4.2, in particular Theorem 4.2.1, which presents a method to solve LQR problems with unknown cost and unknown dynamics directly using data, to solve Problem 5.1.1 and hence determine the Nash equilibrium strategy of player 2 for the considered LQ game with asymmetric information structure. Let the following assumption hold.

Assumption 5.1.2. Player 2 is able to give "exploring" inputs $u_{2,d,[0,T-1]}$ to excite the dynamics (3.1) (with the actions of player 1 fixed) and record the state response $x_{d,[0,T]}$ and performance output response $z_{2,d,[0,T-1]}$.

Consider the data matrices

$$U_{2-} = \begin{bmatrix} u_{2,d}(0) & \dots & u_{2,d}(T-1) \end{bmatrix}, \tag{5.14a}$$

$$Z_{2-} = \begin{bmatrix} z_{2,d}(0) & \dots & z_{2,d}(T-1) \end{bmatrix}, \tag{5.14b}$$

as well as X_- and X_+ as defined in (2.18) with $k_0 = 0$, namely

$$X_- = \begin{bmatrix} x_d(0) & \dots & x_d(T-1) \end{bmatrix},$$
$$X_+ = \begin{bmatrix} x_d(1) & \dots & x_d(T) \end{bmatrix}.$$

In the following result it is shown that the problem of finding the Nash equilibrium strategy of player 2 can be formulated and solved via a purely data-dependent SDP.

Corollary 5.1.3. *Consider Problem 5.1.1 and suppose the available data is such that*

$$\mathit{rank}\left(\begin{bmatrix} X_- \\ U_{2-} \end{bmatrix}\right) = n + m_2.$$

Then, the feedback gain corresponding to the Nash equilibrium strategy $\phi_2^\star(x(k))$ of player 2 is given by

$$K_2^* = U_{2-}H^\star \left(X_- H^\star\right)^{-1}, \tag{5.15}$$

with $H^\star$ the solution of

$$\begin{aligned} \min_{\gamma, S, H, O} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{Tr}\,(O) \leq \gamma, \\ & \begin{bmatrix} S - I_n & X_+ H \\ H^\top X_+^\top & S \end{bmatrix} \succeq 0, \\ & \begin{bmatrix} O & Z_{2-}H \\ H^\top Z_{2-}^\top & S \end{bmatrix} \succeq 0, \\ & X_- H = S. \end{aligned} \tag{5.16}$$

Proof. By Theorem 4.2.1, $u_2(k) = K_2^\star x(k)$ with $K_2^\star$ as given in (5.15) is the solution of the LQR problem defined by the cost (5.10) and the dynamics (5.11). By Proposition 5.1.1, the unique solution of this LQR problem solves Problem 5.1.1. □

While the unknown terms in the cost functional of player 2 and the dynamics perceived by player 2 in Problem 5.1.1 are due to the unknown action of player 1, the data-driven solution proposed in Corollary 5.1.3 also allows player 2 to compensate for a potential lack of knowledge of the system matrices A, B_1 and/or B_2. Similarly, the result only requires samples of the signal z_2 of the form (5.1), the cost weight matrices C_2, D_{21} and D_{22} may or may not be known to player 2. This is highlighted via parentheses in Table 5.1.

Remark 5.1.2. The reformulation of the problem faced by player 2 in the considered class of games with asymmetric information structure in Proposition 5.1.1, and hence the data-driven solution in Corollary 5.1.3 rely on Assumption 5.1.1, which may be difficult to verify or guarantee in practice. Note that if player 1 adheres to a fixed feedback strategy which does not correspond to a Nash equilibrium solution, then the presented methods still result in a control law for player 2 that renders the resulting closed-loop system stable and is optimal with respect to the cost functional of player 2. If player 1 slightly deviates from the Nash equilibrium strategy, the deviation can be considered as process noise. Various direct data-driven methods guaranteeing robustness to noise have been developed, see e.g. [146, 149, 148]. A low-complexity learning framework for LQR controllers from noisy data with stability and performance error guarantees is presented in [156]. While noise is not considered in this section, it is expected that the results from [156] can be applied in a straightforward manner, since the LQR cost function representation proposed in Section 4.2 and utilised Corollary 5.1.3 is not affected by process noise (if the performance output is of the form as defined as in (4.84)).

Two special cases of the considered class of games are explored in the following two remarks.

Remark 5.1.3. In the special case in which the cost of each player does not explicitly depend on the other player's input (i.e. $D_{ij} = 0$, for $i = 1, 2$, $j = 1, 2$ and $j \neq i$), $\bar{C}_2 = C_2$. If C_2, D_{22} are known to player 2, then finding the equilibrium strategy for player 2 reduces to an LQR problem with unknown dynamics matrix $\bar{A}_2$ only, which can be solved by directly applying the result of [95, Theorem 4].

Remark 5.1.4. In the finite-horizon case, feedback Nash equilibrium solutions of LQ discrete-time dynamic games are characterised by the solutions of coupled matrix *difference* equations. The feedback gains corresponding to Nash equilibrium strategies are hence time-varying. Consequently, following a similar reformulation as in Proposition 5.1.1, the unknown cost and system matrices in the resulting LQR problem faced by player 2 are time-varying. This problem can be solved via a data-dependent SDP by extending the data-driven representation for LTV systems introduced in Section 4.1 to parameterise the LQR cost as outlined in Remark 4.2.2.

5.1.3 Example

Consider the interaction dynamics for a contact robot, described by

$$\dot{x} = \begin{bmatrix} 0 & 1 \\ 0 & -J_c^{-1} D_c \end{bmatrix} x + \begin{bmatrix} 0 \\ J_c^{-1} \end{bmatrix} (u_1 + u_2) \, ,$$

with $x = \begin{bmatrix} x_e - x_t & v_e \end{bmatrix}^\top$, where x_e is the end effector position of the robot, x_t is the target position, v_e is the end effector speed and u_1 and u_2 are the force inputs given by the human and the robot, respectively. The inertia and damping coefficients are chosen as $J_c = 6$ kg and $D_c = -0.2$ N/m (as in [20]). The dynamics are discretised using zero-order hold with time step $\Delta = 0.1$ s, which results in a discrete-time system of the form (3.1) with $N = 2$. Consider the task of performing three back and forth arm reaching movements, which involve the human operator guiding the end effector from an initial position to a target position with the support of the robot over a time period of 30 s (i.e. for $k = 0, \dots, 300$). This scenario may be relevant in a rehabilitation setting to train a patient to perform reaching movements, or in a manufacturing setting to support an operator in lifting heavy objects. Inspired by evidence that human behaviour in such interaction settings can be described as Nash equilibrium strategies of a game [215], the human-robot interaction is modelled as a dynamic game (3.1), (5.2), $i = 1, 2$. To this end, assume the human user aims to minimise the cost functional (5.2), $i = 1$, with performance output given by (5.1), $i = 1$, with

$$C_1 = \begin{bmatrix} 10 & 0 \\ 0 & \sqrt{0.1} \\ 0 & 0 \end{bmatrix}, \quad D_{11} = \begin{bmatrix} 0 \\ 0 \\ 0.1 \end{bmatrix}, \quad D_{12} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix},$$

and the contact robot aims to minimise (5.2), $i = 2$, with performance output

$$z_2(k) = \begin{bmatrix} 10(x_e - x_t) \\ 0.3\dot{v}_e \end{bmatrix},$$

i.e. a weighted stacking of the distance to the target position and the end effector acceleration (both signals that are readily available for measurement). Such a control objective encourages driving the end effector to its target position while penalising sudden movements. Noting that $\dot{v}_e \approx \frac{1}{\Delta}(v_e(k+1) - v_e(k))$, the performance output of the robot is of the form (5.1) (with C_2, D_{21} and D_{22} unknown). For the purpose of designing the control input u_2 of the robot, the performance output matrices C_1, D_{11} and D_{12} and the system matrices A, B_1 and B_2 are also considered unknown.

To simulate the human's behaviour, (3.4), (5.3), $i = 1, 2$, and (5.4) are solved numerically to obtain a feedback gain $K_1^\star$ corresponding to a Nash equilibrium solution of the game (3.1), (5.2), $i = 1, 2$, and the human input is fixed as $u_1(k) = K_1^\star x(k)$, for $k = 0, \dots, 300$. To obtain the corresponding Nash equilibrium strategy for u_2, the robot initially provides exploring open-loop inputs, which are sampled randomly from a uniform distribution on the interval $(0, 1)$, for $k = 0, \dots, T$, with $T = 5$, and collects input, state and performance output data. The collected data is then used to populate the matrices U_{2-}, Z_{2-}, X_- and X_+ as defined in (5.14), (2.18), and hence to solve (using CVX [208]) the *data-dependent* SDP (5.16). For $k = T + 1, \dots, 300$, the robot adheres to the obtained strategy $u_2(k) = K_2^\star x(k)$, with $K_2^\star$ given by (5.15), which corresponds to the Nash equilibrium solution $\{K_1^\star x(k), K_2^\star x(k)\}$. The time histories of the states and inputs are shown in Figure 5.1 and Figure 5.2, respectively. It is worth highlighting that only data for $T = 5$ time steps is required to determine the Nash equilibrium strategy for the robot. The data collection interval is negligible compared to the time horizon of the reaching task. In fact, the effects of the exploring inputs given by the robot for $k = 0, \ldots, 5$ on the resulting state trajectories are hardly noticeable.

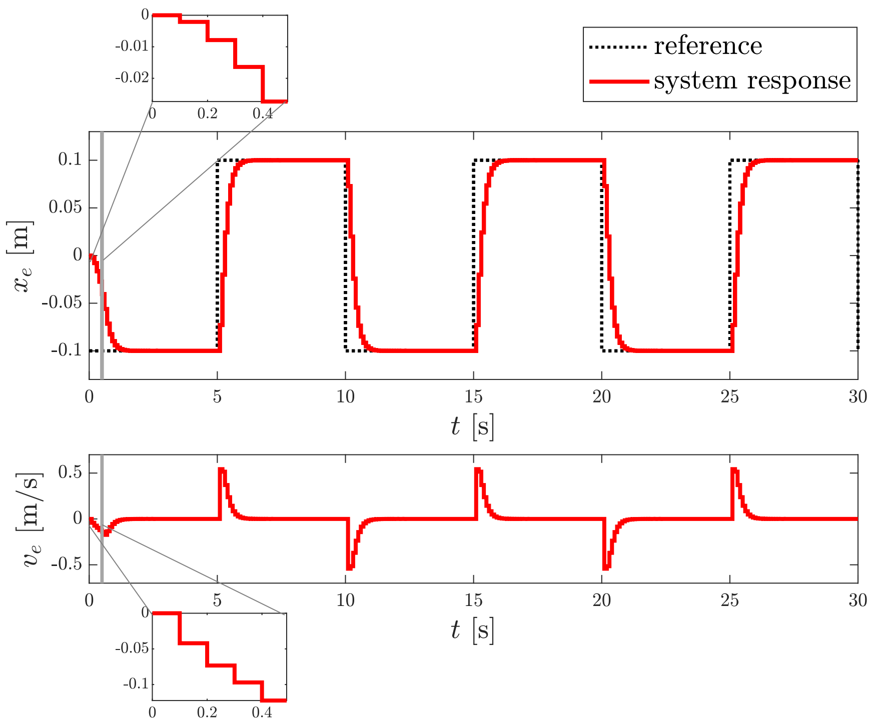


Figure 5.1: Time histories of the system states. The end of the data collection interval is marked by the grey vertical line.

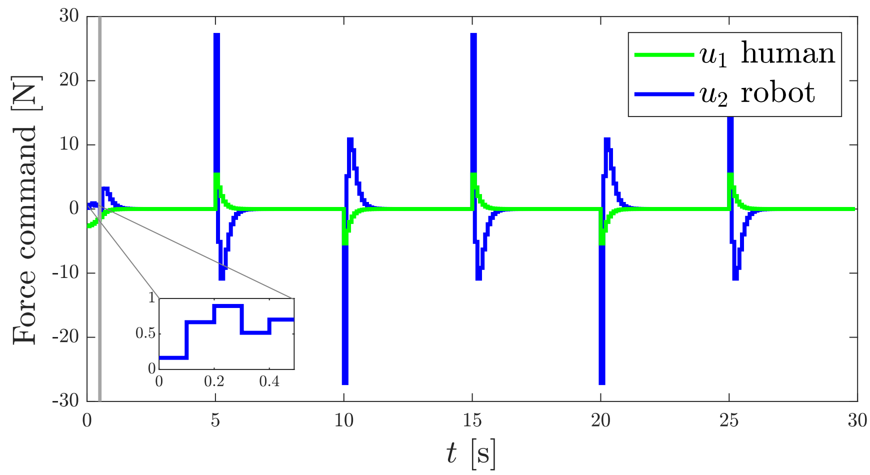


Figure 5.2: Time histories of the inputs. The end of the data collection interval is marked by the grey vertical line.

5.2 Data-driven Nash equilibrium finding algorithms

Motivated by the fact that in various applications of dynamic games it is probable that different players have access to different and potentially incomplete information, Section 5.1 considers a class of LQ discrete-time dynamic games in which one player lacks knowledge regarding the performance criteria the other players are aiming to optimise, and it is proposed how the "uninformed" player can use direct data-driven methods from Section 4.2 to compensate for the lack of information and determine their strategy corresponding to a Nash equilibrium solution of the game. The methods also allow this player to compensate for lack of information regarding the system dynamics.

In this section, the focus instead lies on games in which not one but *all* players are faced with incomplete information. More precisely, assume each player aims to minimise a known objective function, without knowing the objective functions which the other players in the game are aiming to optimise, and the players may not know the system dynamics. Algorithms to determine feedback Nash equilibrium solutions for discrete-time dynamic games with unknown system dynamics have been proposed in [76, 72, 73, 74]. If the system dynamics are known and data corresponding to an equilibrium solution is available, but the objective criteria of some or all players are unknown, then these can be learned or reconstructed using inverse dynamic games methods [81]. In [78] reinforcement learning algorithms are introduced for discrete-time LQ dynamic games involving multi-agent systems, in which the players communicate with their neighbours via a directed graph and only have access to local information. Herein, it is instead demonstrated that the players can jointly converge to a Nash equilibrium solution by scheduling experiments and taking turns to collect finite data sequences of the state and their own input only, without knowledge of each other's objective criteria or of the system dynamics. This is achieved by utilising data-driven methods in a similar way as in Section 5.1 in the context of the iterative Nash equilibrium finding methods introduced in Section 3.4.

The remainder of this section is organised as follows. In Section 5.2.1 the considered problem is introduced. Data-driven algorithms to find Nash equilibria, which are applicable if cost and model information is missing, are proposed in Section 5.2.2. The algorithms are discussed in Section 5.2.3. In Section 5.2.4 the performance of the algorithms is demonstrated and discussed via two illustrative numerical examples, as well as by revisiting the practically motivated example involving human-robot interaction from Section 5.1.3.

5.2.1 Problem formulation

Consider the LQ infinite-horizon discrete-time dynamic game (3.1), (3.2), $i = 1, \ldots, N$. Namely, consider the system

$$x(k+1) = Ax(k) + \sum_{i=1}^{N} B_i u_i(k),$$

with $x \in \mathbb{R}^n$, and let player i aim to minimise the cost functional

$$J_i(x(0), u_1(\cdot), \ldots, u_N(\cdot)) = \sum_{k=0}^{\infty} \left(x(k)^\top Q_i x(k) + \sum_{j=1}^{N} u_j(k)^\top R_{ij} u_j(k) \right),$$

via the choice of control action $u_i \in \mathbb{R}^{m_i}$, for $i = 1, \ldots, N$. Note that this corresponds to the cost (5.2) considered in Section 5.1 in the special case in which $C_i^\top D_{ij} = 0$ and $D_{ij}^\top D_{il} = 0$, for $i = 1, \ldots, N$, $j = 1, \ldots, N$, $l = 1, \ldots, N$, $l \neq j$, which is considered in this section for ease of exposition. Like in the previous section, the focus lies on determining admissible[2] feedback strategies of the form $\phi_i(x(k)) = K_i x(k)$ for the players' actions u_i, $i = 1, \ldots, N$, such that (2.8) holds (see Definition 2.1.2). As shown in Section 3.1, such feedback Nash equilibrium solutions for the considered class of games are characterised by the solutions of a set of coupled algebraic equations, namely (3.5), $i = 1, \ldots, N$, and (3.6), such that the resulting closed-loop system is asymptotically stable, i.e. (3.4) holds (see Theorem 3.1.1). However, as discussed in Chapter 3, determining $K_i^\star$ and $P_i^\star$, for $i = 1, \ldots, N$, which satisfy (3.4), (3.5), $i = 1, \ldots, N$, and (3.6), and thereby determining feedback Nash equilibrium solutions of the game (3.1), (3.2), $i = 1, \ldots, N$, is generally challenging. In Section 3.4 this is addressed by proposing iterative methods, which involve the solution of matrix equations of reduced complexity with respect to (3.5), $i = 1, \ldots, N$, and (3.6) at each iteration step. In the presence of complete model and cost information, these methods allow to iteratively determine a feedback Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \ldots, N$. Motivated by scenarios in which it is likely that different and typically partial information is available to each player in the game, the focus of this section lies on developing iterative methods for games with incomplete information. More precisely, the case is considered in which each player i only knows the cost matrices corresponding to the cost J_i (as given in (3.2)) player i is aiming to minimise, but player i does not know the cost matrices corresponding to the cost functionals J_j for $j = 1, \ldots, N$, $j \neq i$, associated with all the other players, and the players may not know the system matrices A and B_i, $i = 1, \ldots, N$, as formalised in the following statement.

Problem 5.2.1. Consider the game (3.1), (3.2), $i = 1, \ldots, N$. Let the cost matrices Q_w and R_{wg}, for $g = 1, \ldots, N$, associated with players w, for $w = 1, \ldots, N$, $w \neq i$, and the system matrices A, B_j, for $j = 1, \ldots, N$, be unknown[3] to player i, for $i = 1, \ldots, N$. Determine a Nash equilibrium solution of the game.

Problem 5.2.1 can be considered a more general case of Problem 5.1.1 considered in Section 5.1, in which not one but *all* players in the game do not know the cost matrices associated with the other players and do not know the system dynamics. The problem is addressed by presenting methods

[2] In the sense of Definition 3.3.1.

[3] Note that the considered problem is interesting and relevant even if the system matrices are known to some or all of the players and only the cost matrices of the respective other players are unknown to each player. However, the presented results also allow to account for unknown system dynamics. To consider the most general case the system dynamics are hence treated as unknown.

to iteratively determine gains $K_i^\star$, for $i = 1, \ldots, N$, corresponding to a feedback Nash equilibrium solution of the game (3.1), (3.2), $i = 1, \ldots, N$, starting from an initial guess $K_i^{(0)}$, for $i = 1, ..., N$. The iterative updates constitute a data-driven version of the algorithms presented in Section 3.4 and involve each player i using data of the state x and the own input u_i collected via sequential experiments to update the strategy choice $K_i^{(l)}$ at each iteration (l), $l \in \mathbb{N}$.

5.2.2 Data-driven algorithms for Nash equilibria

Towards designing data-driven methods to iteratively converge to a Nash equilibrium solution of an LQ dynamic game despite each player having only limited information regarding the opponents' performance criteria and system dynamics consider the following assumption.

Assumption 5.2.1. The signals $x(k)$ and $u_i(k)$, are available for measurement for player i, $i = 1, \ldots, N$. The players are able and willing to schedule experiments, taking turns to recursively collect sequences of data. During the turn of player i to collect data with the aim of updating the strategy guess from $K_i^{(l)}$ to $K_i^{(l+1)}$, the player collects[4] the state-response $x_{d,[k_0,k_0+T_i^{(l+1)}]}$ to the sequence of "exploring" inputs $u_{i,d,[k_0,k_0+T_i^{(l+1)}-1]}$, assuming all other players, $j = 1, \ldots, N$, $j \neq i$, stick to a constant state-feedback strategy corresponding to their current guess of the Nash equilibrium strategy (i.e. at iteration $(l+1)$, $u_w(k) = K_w^{(l+1)}x(k)$, for $w = 1, \ldots, i-1$ and $u_j(k) = K_j^{(l)}x(k)$, for $j = i+1, \ldots, N$, for $k = k_0, \ldots, k_0 + T_j^{(l+1)} - 1$).

At each iteration (l) and for each player i, the data is arranged to form the matrices

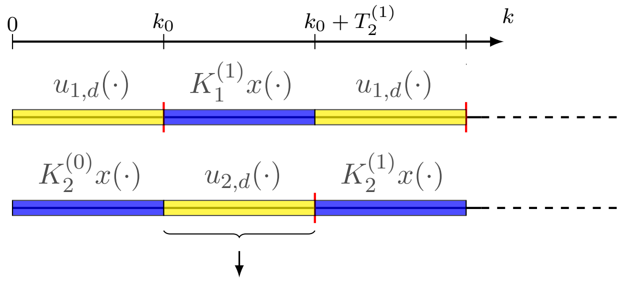
$$U_{i-} = U_{i-}^{(l)} = \begin{bmatrix} u_{i,d}(k_0) & \ldots & u_{i,d}(k_0 + T_i^{(l)} - 1) \end{bmatrix}, \tag{5.17a}$$

$$X_{-} = X_{i-}^{(l)} = \begin{bmatrix} x_d(k_0) & \ldots & x_d(k_0 + T_i^{(l)} - 1) \end{bmatrix}, \tag{5.17b}$$

$$X_{+} = X_{i+}^{(l)} = \begin{bmatrix} x_d(k_0 + 1) & \ldots & x_d(k_0 + T_i^{(l)}) \end{bmatrix}. \tag{5.17c}$$

For notational convenience, the subscripts and superscripts indicating the dependence on the player i (for the state matrices) and the iteration (l) (for both the state and input matrices) are omitted, i.e. the matrices containing the data collected by player i are denoted by U_{i-}, X_- and X_+ at each iteration (l), for $i = 1, \ldots, N$ and $l \in \mathbb{N}$. Similarly, the starting time instance of each experiment, for each player i and at each iteration (l), is denoted by k_0. The scheduling of data collection and strategy updates described in Assumption 5.2.1 is illustrated in Figure 5.3 for the case $N = 2$. Note that the update scenario described in Assumption 5.2.1, more precisely the current strategy guesses at which players j, $j = 1, \ldots, N$, $j \neq i$, fix their strategies during the update of player i, $i = 1, \ldots, N$, corresponds to the *asynchronous* version of the algorithms presented in Section 3.4. This is chosen

[4]The subscripts indicate the sequences correspond to data measured by player i at iteration $(l+1)$, over the interval $[k_0, k_f]$, with $k_f = k_0 + T_i^{(l+1)}$ or $k_f = k_0 + T_i^{(l+1)} - 1$ as appropriate, where $T_i^{(l+1)}$ is the length of the data collection experiment. As in previous chapters and sections the subscript d indicates measured data samples.



$$\begin{aligned}
U_{2-} &= \begin{bmatrix} u_{2,d}(k_0) & \dots & u_{2,d}(k_0 + T_2^{(1)} - 1) \end{bmatrix} \\
X_{-} &= \begin{bmatrix} x_d(k_0) & \dots & x_d(k_0 + T_2^{(1)} - 1) \end{bmatrix} \\
X_{+} &= \begin{bmatrix} x_d(k_0 + 1) & \dots & x_d(k_0 + T_2^{(1)}) \end{bmatrix}
\end{aligned}$$

Figure 5.3: Illustration of the experiment scheduling with $N = 2$ players. The yellow blocks indicate phases in which a player gives exploring inputs to excite the system dynamics and collects data, while the blue blocks indicate phases in which a player applies a constant feedback action. The instances at which the players update their strategy guesses are highlighted by the red vertical lines.

since it seems natural in practice for each player to update their strategy immediately after collecting data, rather than waiting for all players to finish their experiments. However, the synchronous version could be implemented using data following analogous steps.

With the information structure specified in Problem 5.2.1, player i does not have enough information to solve the coupled algebraic matrix equations (3.5), $i = 1, \dots, N$, and (3.6), for a solution such that (3.4) holds, nor to apply the model-based Nash equilibrium finding algorithms introduced in Section 3.4. If Assumption 5.2.1 holds, then during the strategy update of player i the strategies of the other players are fixed at state-feedback strategies. Hence, the dynamics perceived by player i when updating the strategy guess from iteration (l) to $(l+1)$ are

$$x(k+1) = \hat{A}_{a,i}^{(l+1)} x(k) + B_i u_i(k), \tag{5.18}$$

with $\hat{A}_{a,i}^{(l+1)} = A + \sum_{w=1}^{i-1} B_w K_w^{(l+1)} + \sum_{j=i+1}^{N} B_j K_j^{(l)}$ as defined in Section 3.4. That is, the current guesses of the Nash equilibrium strategies for all other players j, $j = 1, \dots, N$, $j \neq i$, which depend on their cost function weights Q_j, R_{jw}, $w = 1, \dots, N$, and are unknown to player i, are encapsulated in the dynamics matrix $\hat{A}_{a,i}^{(l+1)}$. The lack of knowledge is overcome in this section by recovering or representing the perceived dynamics using data, utilising the indirect and direct data-driven methods introduced in [95].

Remark 5.2.1. Note that the presented approaches, which represent the dynamics perceived by player i, $i = 1, \dots, N$, containing the unknown actions of the other players j, $j = 1, \dots, N$, $j \neq i$, using data also allow to account for lack of knowledge of the system dynamics. To address the most general case, A and B_i, for $i = 1, \dots, N$, are considered unknown, as specified in Problem 5.2.1. Note, however, that the presented results are equally relevant if the system matrices are known. If faced with the alternative problem in which the cost weight matrices of all players are known to each player and only the system dynamics are unknown, then assuming the inputs of all players are available for measurement, the data-driven methods from [95] can be combined with the algorithms from Section 3.4 of this thesis to design alternative centralised data-driven algorithms, which do not require scheduled experiments as considered herein and are hence more sample efficient. This described approach could be considered as an alternative to the methods proposed in [72, 73, 74], however, this is not the focus of this section.

Finally, for ease of exposition consider the following assumption throughout this section.

Assumption 5.2.2. The cost functional (3.2), which player i, $i = 1, \dots, N$, aims to minimise, is such that $R_{ij} = 0$, for $j = 1, \dots, N$, $j \neq i$.

However, the data-driven results can be extended to include cost weight terms $R_{ij} \neq 0$ in a straightforward manner by following analogous steps, at the cost of more cumbersome notation and the requirement to collect additional data of a performance variable, as detailed in Section 5.1.

As in Section 3.4, two types of update law are presented: the "Lyapunov iterations" involving the solution of Lyaponov equations and the "Riccati iterations" involving the solution of Riccati equations.

Lyapunov iterations

Consider the iterative update law (3.72) with $\sigma = a$, namely

$$0 = \hat{Q}_{a,i}^{(l+1)} + K_i^{(l)\top} R_{ii} K_i^{(l)} - P_i^{(l+1)} + \left(\hat{A}_{a,i}^{(l+1)} + B_i K_i^{(l)}\right)^\top P_i^{(l+1)} \left(\hat{A}_{a,i}^{(l+1)} + B_i K_i^{(l)}\right), \tag{5.19a}$$

$$K_i^{(l+1)} = -\left(R_{ii} + B_i^\top P_i^{(l+1)} B_i\right)^{-1} B_i^\top P_i^{(l+1)} \hat{A}_{a,i}^{(l+1)}, \tag{5.19b}$$

for $i = 1, \dots, N$ and $l \in \mathbb{N}$. In Algorithm 1 a data-driven implementation of (5.19) is introduced, which is proposed as a solution to Problem 5.2.1 in the following result.

Proposition 5.2.1. *Consider Problem 5.2.1 and let Assumption 5.2.1 and Assumption 5.2.2 hold. Suppose the collected data for player i is such that the matrices U_{i-}, X_- and X_+ as defined in* (5.17) *satisfy*

$$rank\left(\begin{bmatrix} X_- \\ U_{i-} \end{bmatrix}\right) = n + m_i, \tag{5.20}$$

for $i = 1, \dots, N$, at each iteration (l). Then, if the set of feedback gains $\{K_1^\star, \dots, K_N^\star\}$ obtained via Algorithm 1 is such that (3.4) *holds and the conditions stated in Proposition 3.4.1 hold, Algorithm 1 solves Problem 5.2.1.*

Proof. If Assumptions 5.2.1 and 5.2.2 hold, then $\hat{Q}_{a,i}^{(l+1)} = Q_i$ and if (5.20) holds then by [95, Theorem 1] the dynamics identified in STEPS 19-23 of Algorithm 1 are such that $(\hat{A}_i, \hat{B}_i) = (\hat{A}_{a,i}^{(l+1)}, B_i)$, for $i = 1, \ldots, N$, at each iteration (l). Hence, the updates in STEP 24 and STEP 25 are equivalent to (5.19). Convergence to a feedback Nash equilibrium follows directly from Proposition 3.4.1. □

Remark 5.2.2. STEPS 19-23 in Algorithm 1 correspond to identifying the dynamics perceived by player i, $i = 1, \ldots, N$, at iteration $(l+1)$ due to the unknown state-feedback actions of the other players j, $j = 1, \ldots, N$, $j \neq i$. While the indirect data-driven method from [95, Theorem 1] is used herein, the steps could also be replaced with alternative system identification techniques.

Algorithm 1 - Data-driven Lyapunov iterations

1: **Initialise:** $x(0) = x_0$, $l = 0$, $k = 0$, $k_0 = k$, $\varepsilon = 1$
2: **Specify:** tolerance $\bar{\varepsilon}$, stabilising $K_j^{(0)}$, for $j = 1, \ldots, N$, and time horizon T_f
3: **while** $\varepsilon > \bar{\varepsilon}$ **do**
4: **for** $i = 1$ to N **do**
5: **Assume:** $u_w(k) = K_w^{(l+1)} x(k)$, for $w = 1, \ldots, i-1$, and $u_j(k) = K_j^{(l)} x(k)$, for $j = i+1, \ldots, N$
6: **Data collection:**
7: clear X_-, X_+, U_{i-}
8: **while** $\operatorname{rank}\left(\begin{bmatrix} X_-^\top & U_{i-}^\top \end{bmatrix}^\top\right) < n + m_i$ **do**
9: give exploring input $u_{i,d}(k)$
10: measure $x_d(k)$
11: $X_- = \begin{bmatrix} x_d(k_0) & \ldots & x_d(k) \end{bmatrix}$
12: $U_{i-} = \begin{bmatrix} u_{i,d}(k_0) & \ldots & u_{i,d}(k) \end{bmatrix}$
13: $k \leftarrow k + 1$
14: **end while**
15: measure $x_d(k)$
16: $X_+ = \begin{bmatrix} x_d(k_0 + 1) & \ldots & x_d(k) \end{bmatrix}$
17: $k_0 \leftarrow k$
18: **Policy update:**
19: **if** $l = 0$ **then**
20: compute $\begin{bmatrix} \hat{A}_i & \hat{B}_i \end{bmatrix} = X_+ \left(\begin{bmatrix} X_-^\top & U_{i-}^\top \end{bmatrix}^\top\right)^\dagger$
21: **else**
22: compute $\hat{A}_i = \left(X^+ - \hat{B}_i U_{i-}\right) X_-^\dagger$
23: **end if**
24: $P_i^{(l+1)} = Q_i + K_i^{(l)\top} R_{ii} K_i^{(l)} + \left(\hat{A}_i + \hat{B}_i K_i^{(l)}\right)^\top P_i^{(l+1)} \left(\hat{A}_i + \hat{B}_i K_i^{(l)}\right)$
25: $K_i^{(l+1)} = -\left(R_{ii} + \hat{B}_i^\top P_i^{(l+1)} \hat{B}_i\right)^{-1} \hat{B}_i^\top P_i^{(l+1)} \hat{A}_i$
26: let $u_i(k) = K_i^{(l+1)} x(k)$
27: **end for**
28: $l \leftarrow l + 1$
29: $\varepsilon \leftarrow \max_i \left(\left\| K_i^{(l)} - K_i^{(l-1)} \right\|\right)$
30: **end while**
31: $K_i^\star = K_i^{(l)}$, for $i = 1, \ldots, N$
32: **while** $k < T_f$ **do**
33: let $u_i(k) = K_i^\star x(k)$, for $i = 1, \ldots, N$
34: $k \leftarrow k + 1$
35: **end while**

Remark 5.2.3. Note that being based on the Lyapunov iterations (3.72), $\sigma = a$, from Section 3.4, Algorithm 1 requires a stabilising set of gains $\left\{K_1^{(0)}, \dots, K_N^{(0)}\right\}$ as initial guess. A possible strategy to compute stabilising $K_i^{(0)}, i = 1, \dots, N$, in a data-driven framework consists in selecting arbitrary[5] $K_j^{(0)}$, for $j = 1, \dots, N-1$, and in letting player N, without loss of generality, perform, in advance and only once as a preliminary initialisation, STEPS 5 - 25 of Algorithm 1. As a consequence, the computed $K_N^{(0)}$ has the property that $\rho\left(\hat{A}_N + \hat{B}_N K_N^{(0)}\right) < 1$, and hence the selection of the initial control gains is (collectively) stabilising the closed-loop system.

Riccati iterations

Consider the iterative update law (3.75) with $\sigma = a$, namely

$$0 = \hat{Q}_{a,i}^{(l+1)} + K_i^{(l+1)^\top} R_{ii} K_i^{(l+1)} - P_i^{(l+1)} + \left(\hat{A}_{a,i}^{(l+1)} + B_i K_i^{(l+1)}\right)^\top P_i^{(l+1)} \left(\hat{A}_{a,i}^{(l+1)} + B_i K_i^{(l+1)}\right), \tag{5.21a}$$

$$K_i^{(l+1)} = -\left(R_{ii} + B_i^\top P_i^{(l+1)} B_i\right)^{-1} B_i^\top P_i^{(l+1)} \hat{A}_{a,i}^{(l+1)}, \tag{5.21b}$$

for $i = 1, \dots, N$ and $l \in \mathbb{N}$. To formulate a data-driven equivalent to (5.21) recall the following result, which introduces a method to design optimal controllers (for LTI systems influenced by the action of a single decision maker, with the aim of minimising a quadratic cost function) directly using data, without requiring knowledge of the system dynamics.

Lemma 5.2.1. *[95, Theorem 4] Consider system* (2.17)*, namely*

$$x(k+1) = Ax(k) + Bu(k),$$

with $x \in \mathbb{R}^n$ and $u \in \mathbb{R}^m$, and assume input-state data is available to form the matrices U_-, X_-, and X_+ as defined in (2.18)*, such that* (2.19) *holds, i.e.*

$$\mathit{rank}\left(\begin{bmatrix} X_- \\ U_- \end{bmatrix}\right) = n + m.$$

The optimal state-feedback control gain $K^\star$, such that $u(k) = K^\star x(k)$ minimises the cost function

$$J(x(0), u(\cdot)) = \sum_{k=0}^{\infty} \left(x(k)^\top Q x(k) + u(k)^\top R u(k)\right), \tag{5.22}$$

with $R \succ 0$ and $Q \succeq 0$, is given by $K^\star = U_- G^\star$, where $G^\star = H^\star S^{\star -1}$, with $H^\star$ and $S^\star$ a solution of

[5]The selected arbitrary strategies still need to be such that the pair $(A + \sum_{i=1}^{N-1} B_i K_i^{(0)}, B_N)$ is stabilisable. While this may be difficult to verify if the system dynamics are unknown, if there exists a feedback Nash equilibrium to the game (3.1), (3.2), $i = 1, \dots, N$, then such a set of strategies exists.

the convex optimisation problem

$$\begin{aligned}
\min_{\gamma, S, H, O} \quad & \gamma \\
\text{s.t.} \quad & \mathbf{Tr}\,(QS) + \mathbf{Tr}\,(O) \leq \gamma, \\
& \begin{bmatrix} S - I_n & X_+ H \\ H^\top X_+^\top & S \end{bmatrix} \succeq 0, \\
& \begin{bmatrix} O & R^{\frac{1}{2}} U_- H \\ H^\top U_-^\top R^{\frac{1}{2}} & S \end{bmatrix} \succeq 0, \\
& X_- H = S.
\end{aligned} \tag{5.23}$$

As a consequence of Lemma 5.2.1, solving the data-driven optimisation problem (5.23) is equivalent to solving the algebraic Riccati equation

$$0 = Q + K^{\star\top} R K^\star - P^\star + (A + BK^\star)^\top P^\star (A + BK^\star), \tag{5.24a}$$

with

$$K^\star = -\left(R + B^\top P^\star B\right)^{-1} B^\top P^\star A, \tag{5.24b}$$

associated with the LQR problem defined by the cost (5.22) subject to the dynamics (2.17). By construction, Lemma 5.2.1 assumes $P^\star \succ 0$, hence consider the following assumption.

Assumption 5.2.3. The coupled algebraic matrix equations (3.5), $i = 1, \ldots, N$, and (3.6) associated with the game (3.1), (3.2), $i = 1, \ldots, N$, admit a solution $\{K_1^\star, \ldots, K_N^\star\}$, $\{P_1^\star, \ldots, P_N^\star\}$ such that $P_i^\star \succ 0$, for $i = 1, \ldots, N$.

In the following result, Algorithm 2, which utilises Lemma 5.2.1, is proposed as a solution to Problem 5.2.1.

Proposition 5.2.2. *Consider Problem 5.2.1 and let Assumption 5.2.1, Assumption 5.2.2 and Assumption 5.2.3 hold. Suppose the collected data for player i is such that the matrices U_{i-}, X_- and X_+ as defined in* (5.17) *satisfy* (5.20)*, for $i = 1, \ldots, N$, at each iteration* (l)*. Then, if the set of feedback gains $\{K_1^\star, \ldots, K_N^\star\}$ obtained via Algorithm 2 is such that the conditions stated in Proposition 3.4.2 hold, Algorithm 2 solves Problem 5.2.1.*

Proof. Note that (5.24) is equal to (5.21) with $A = \hat{A}_{a,i}^{(l+1)}$, $B = B_i$, $Q = \hat{Q}_{a,i}^{(l+1)}$, $R = R_{ii}$, $P^\star = P_i^{(l+1)}$ and $K^\star = K_i^{(l+1)}$. If Assumptions 5.2.1, 5.2.2, and 5.2.3 hold, then $\hat{Q}_{a,i}^{(l+1)} = Q_i$ and if (5.20) holds then by Lemma 5.2.1, STEP 9 and STEP 10 of Algorithm 2 are equivalent to (5.21). Convergence to a feedback Nash equilibrium follows directly from Proposition 3.4.2. □

Algorithm 2 - Data-driven Riccati Iterations

1: **Initialise:** $x(0) = x_0$, $l = 0$, $k = 0$, $k_0 = k$, $\varepsilon = 1$
2: **Specify:** tolerance $\bar{\varepsilon}$, $K_j^{(0)}$, for $j = 2, \dots, N$, and time horizon T_f
3: **while** $\varepsilon > \bar{\varepsilon}$ **do**
4: **for** $i = 1$ to N **do**
5: **Assume:** $u_w(k) = K_w^{(l+1)} x(k)$, for $w = 1, \dots, i-1$, and $u_j(k) = K_j^{(l)} x(k)$, for $j = i+1, \dots, N$
6: **Data collection:**
7: follow STEPS 7 - 17 of Algorithm 1
8: **Policy update:**
9: solve (5.23) with $Q = Q_i$, $R = R_{ii}$, $U_- = U_{i-}$ for $H^\star$, $S^\star$
10: $K_i^{(l+1)} = U_{i-} H^\star S^{\star -1}$
11: let $u_i(k) = K_i^{(l+1)} x(k)$
12: **end for**
13: $l \leftarrow l + 1$
14: $\varepsilon \leftarrow \max_i \left(\left\| K_i^{(l)} - K_i^{(l-1)} \right\| \right)$
15: **end while**
16: $K_i^\star = K_i^{(l)}$, for $i = 1, \dots, N$
17: follow STEPS 32 - 35 of Algorithm 1

Remark 5.2.4. Note that the right hand side of STEP 10 in Algorithm 2 does not explicitly depend on $K_i^{(l)}$ or the strategies of the other players as specified in STEP 5. This dependency is given implicitly via the data matrices X_+, X_- and U_{i-} as defined in (5.17) used to solve (5.23), which are repopulated for each player i at each iteration (l).

5.2.3 Discussion

Algorithms 1 and 2 constitute data-driven versions of the iterative Nash equilibrium finding methods introduced in Section 3.4, which make it possible to overcome the limited information available to each player in Problem 5.2.1. Note that an *indirect* data-driven method is used in Algorithm 1 and a *direct* data-driven method is used in Algorithm 2. The indirect (system identification) method can be readily applied in combination with any model-based control technique and is computationally cheaper for noise-free linear systems as considered herein. In the given setting, the indirect data-driven method further has the advantage that B_i only needs to be identified once for each player i, $i = 1, \dots, N$, at the first iteration (see STEPS 19-23 of Algorithm 1). Thus, the indirect method allows to incorporate available partial system knowledge in a more straightforward way than the direct data-driven method, in which the closed-loop dynamics are entirely represented using data. While the indirect method could be used in combination with both the Lyapunov iterations (3.72) and the Riccati iterations (3.75), it is chosen in this section only for the former. This choice is motivated by the fact that the structure of the Riccati iterations (3.75) allows to readily apply the recent direct data-driven results introduced in [95]. Such methods have potential for systems for which system identification is difficult or involved [95]. Hence, the presented results may serve as a basis for future work on games involving more complicated systems.

Both Algorithm 1 and Algorithm 2 possess an appealing feature which has recently become particularly desirable in modern applications, especially for problems involving a large number of players. Namely, if the algorithms converge, a Nash equilibrium solution is obtained based solely on the partial information available to each player. The presented data-driven algorithms are hence *distributed* in the sense that player i only solves the own matrix equations (3.5) and the i^{th} block row of (3.6), by relying only on measured input-state data, even though these equations belong to a system of coupled matrix equations, for $i = 1, \ldots, N$. No knowledge of the cost matrices associated with the other players nor the system matrices is required. Apart from the scheduling of experiments, no information exchange between the players is required, i.e. player i does not know the strategy guesses of the other players j, $j = 1, \ldots, N$, $j \neq i$, at each iteration, nor does player i measure their inputs u_j, for $i = 1, \ldots, N$, $j = 1, \ldots, N$, $j \neq i$. By capturing the actions of the other players in the dynamics (5.18) and replacing these with data, player i only updates strategy $K_i^{(l)}$ at each iteration (l) and does not need to explicitly estimate the strategies of the other players. Another benefit of this approach is that the computational complexity of the algorithms for each player i does not depend on the total number of players N in the game. More precisely, at each iteration step (l) player i collects $T_i^{(l)} + 1$ samples of the state response to the own $T_i^{(l)}$ exploring inputs, where $T_i^{(l)}$ is such that the rank condition (5.20) holds, which implies $T_i^{(l)} \geq n + m_i$. Hence, if the exploring input signal is chosen well, only $n + m_i + 1$ data points are required per iteration for player i, for $i = 1, \ldots, N$. The total number of data points a player needs to collect depends on the number of iterations until convergence to the specified tolerance, which in turn depends on the system and cost parameters of the specific problem and the chosen initial conditions as illustrated for the model-based version of the algorithms in Section 3.4.4. In the case of Algorithm 2 the accuracy of the chosen solver for the SDP (5.23) in the update step (STEP 9) may also effect the number of iterations until convergence. In Algorithm 1 the update law at each iteration involves the solution of two linear matrix equations for the system identification step (STEP 19 or STEP 23) and the policy evaluation step (STEP 24), respectively. The former is of dimension $n \times T_i^{(l)}$ with $n(n+m_i)$ unknowns if $l = 0$ and n^2 unknowns if $l > 0$. The latter is of dimension $n \times n$ with $(n^2 + n)/2$ unknowns. In Algorithm 2, system representation and policy update at each iteration are combined (see STEPS 9 and 10) and involve the solution of a convex programme with two LMI constraints of dimension $2n \times 2n$ and $(n + m_i) \times (n + m_i)$ and a linear equality constraint of dimension $n \times n$ with a total number of $T_i^{(l)} n + (n^2 + n)/2 + (m_i^2 + m_i)/2$ decision variables.

Finally, recall that Algorithm 1 is applicable for games for which Assumption 5.2.3 does not hold. However, Algorithm 2 (which relies on Assumption 5.2.3) allows a non-stabilising initial guess $K_j^{(0)}$, for $j = 2, \ldots, N$, which may be beneficial in some scenarios, particularly in the context of unknown system dynamics.

5.2.4 Examples

The performance of the presented algorithms is demonstrated via two illustrative numerical examples, before the human-robot interaction example from Section 5.1.3 is revisited. The first numerical example involves a scalar four-player game, which illustrates the experiment scheduling and the efficacy of the algorithms. The second numerical example considers a two-player game involving a slightly larger state-dimension $n = 3$ and allows to put the data requirements of the presented algorithms into perspective. The final example considers the interaction between a human operator and a contact robot and demonstrates the practical relevance of the presented results.

Scalar four-player example

To illustrate the efficacy of the proposed Algorithms 1 and 2, consider the scalar numerical example described by system (3.1), with $N = 4$, and parameters

$$A = 4.83,\ B_1 = 0.54,\ B_2 = 0.97,\ B_3 = 0.71,\ B_4 = 0.69,$$

and the cost functionals (3.2), $i = 1, \dots, 4$, with

$$\begin{aligned} Q_1 &= 0.21, \quad & Q_2 &= 0.006, \quad & Q_3 &= 0.43, \quad & Q_4 &= 0.19, \\ R_{11} &= 0.97, \quad & R_{22} &= 0.25, \quad & R_{33} &= 0.78, \quad & R_{44} &= 0.86, \end{aligned}$$

and in line with Assumption 5.2.2 $R_{12} = R_{13} = R_{14} = R_{21} = R_{23} = R_{24} = R_{31} = R_{32} = R_{34} = R_{41} = R_{42} = R_{43} = 0$. Considering the information structure described in Problem 5.2.1, let the initial set of gains be $K_j^{(0)} = 0$, for $j = 1, 2, 3$ and $K_4^{(0)} = -7$ and note that (3.1) in closed-loop under state-feedback with these initial gains is asymptotically stable. Starting from $x_0 = 0.98$, the players take turns to give exploring inputs, while all other players stick to their current strategy guesses, and collect data to update their strategies using Algorithms 1 and 2, respectively. The exploring inputs are sampled uniformly from the interval $(-1, 1)$. The corresponding time histories of the state and the input of player 1 are shown in Figure 5.4. The dotted grey vertical lines indicate the start of each new experiment for data collection. Both Algorithms 1 and 2 converge with tolerance $\bar{\varepsilon} = 10^{-5}$ within 72 time steps, as indicated by the solid grey vertical lines, which corresponds to 9 iterations of the algorithms. For the remainder of the time horizon ($T_f = 100$) the players follow their determined equilibrium strategies, characterised by the state-feedback gains

$$K_1^\star = -0.026, \quad K_2^\star = -0.005, \quad K_3^\star = -0.087 \quad \text{and} \quad K_4^\star = -6.580,$$

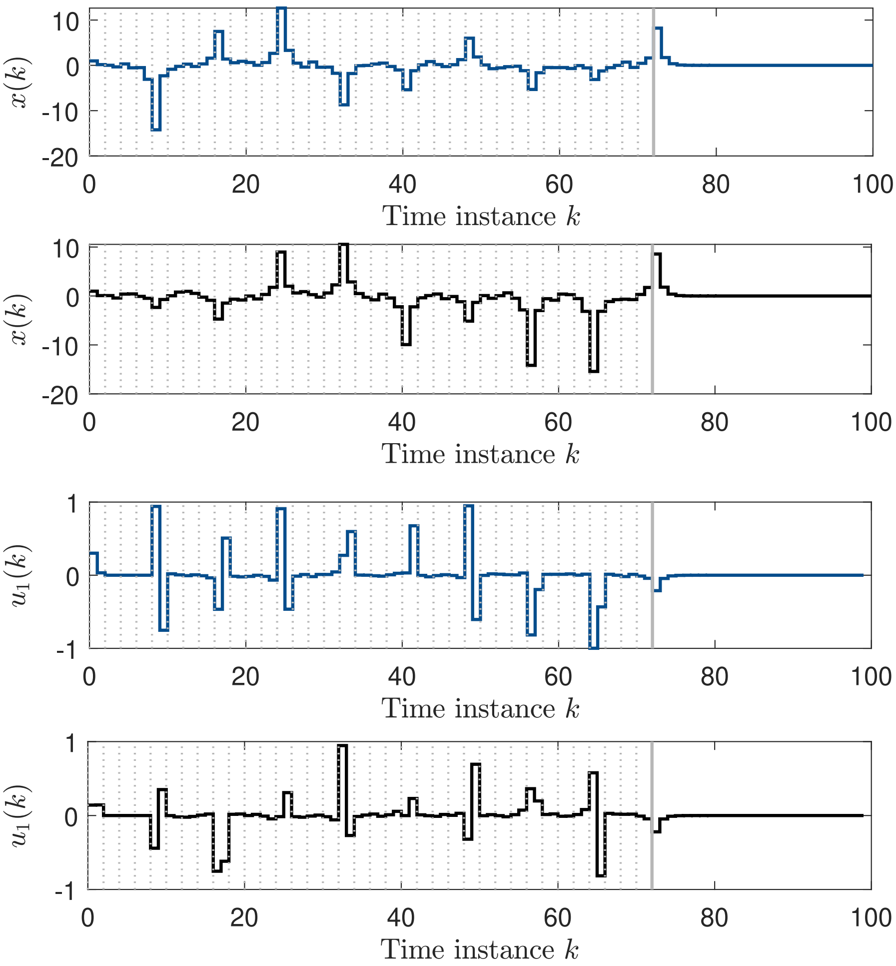


Figure 5.4: Time histories of the state and the input of player 1 using Algorithm 1 (dark blue) and Algorithm 2 (black). The dotted grey lines indicate the start of each new experiment, whereas the solid grey line indicates the end of the scheduled experiments.

which satisfy (3.4), (3.5), $i = 1, \dots, 4$, and (3.6) with

$$P_1^\star = 0.220, \quad P_2^\star = 0.006, \quad P_3^\star = 0.456 \quad \text{and} \quad P_4^\star = 39.138,$$

where all values are rounded to three decimal places.

Comparison of data requirements

While there are several data-driven methods for the considered class of LQ discrete-time dynamic games which consider unknown system dynamics [72, 73, 74] or different local information [78], the author is not aware of any learning algorithms considering the same underlying information structure as specified in Problem 5.2.1 and hence allowing a fair comparison. However, to put the data requirements of the proposed algorithms into perspective, consider the game defined by the system dynamics (3.1) with the system matrices from the example in [73, Section V.C], namely

$$A = \begin{bmatrix} 0.9065 & 0.0816 & -0.0005 \\ 0.0743 & 0.9012 & -0.0007 \\ 0 & 0 & 0.1327 \end{bmatrix}, \quad B_1 = \begin{bmatrix} -0.0027 \\ -0.0068 \\ 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 1 \\ 0.0062 \\ 0 \end{bmatrix},$$

and the cost functionals (3.2), $i = 1, 2$, with

$$Q_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; Q_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \; R_{11} = 1, \; R_{22} = 1, \; R_{12} = R_{21} = 0.$$

Consider the information structure described in Problem 5.2.1 and initial conditions $K_1^{(0)} = \begin{bmatrix} 0 & -0.12 & -1 \end{bmatrix}$, $K_2^{(0)} = \begin{bmatrix} -1 & -0.5 & 0 \end{bmatrix}$, which render the system (3.1) asymptotically stable, and $x_0 = \begin{bmatrix} 10 & -10 & -3 \end{bmatrix}^\top$. The exploring inputs used by both players during their respective data collection phases are sampled uniformly from the interval $(-1, 1)$. Algorithm 1 converges to the tolerance $\bar{\varepsilon} = 10^{-5}$ to the feedback Nash equilibrium characterised by

$$K_1^\star = \begin{bmatrix} 0.0023 & 0.0199 & -0.0666 \end{bmatrix}, \quad K_2^\star = \begin{bmatrix} -0.6444 & -0.8735 & 0.0004 \end{bmatrix},$$

and

$$P_1^\star = \begin{bmatrix} 1.1198 & 0.2434 & -0.0001 \\ 0.2434 & 6.8645 & -0.0016 \\ -0.0001 & -0.0016 & 1.0088 \end{bmatrix}, \quad P_2^\star = \begin{bmatrix} 1.7611 & 2.3731 & -0.0007 \\ 2.3731 & 27.8101 & -0.0048 \\ -0.0007 & -0.0048 & 1.0044 \end{bmatrix},$$

where all values are rounded to four decimal places, and which satisfy (3.4), (3.5), $i = 1, 2$, and (3.6), in 4 iterations and Algorithm 2 in only 3 iterations. With the chosen exploring inputs, each player i, $i = 1, 2$, requires $n + m_i + 1 = 5$ data samples per iteration. Hence, each player collects $5 \times 4 = 20$ data samples until Algorithm 1 converges, and respectively $5 \times 3 = 15$ data samples until Algorithm 2 converges. In comparison [73, Algorithm 4], which only considers the system dynamics to be unknown, requires $N_d \geq n^2 + m_1 n + m_1^2 + m_1 m_2 + m_2 n + m_2 m_1 + m_2^2 = 19$ data samples, before the learning phase of the algorithm even begins.

Practical example: human-robot interaction

Consider the discretised interaction dynamics of a contact robot and a human operator described in Section 5.1.3. The considered task involves two arm reaching movements of the human operator to guide the robot's end effector from an initial position to a target position, supported by the robot over a time period of 20 s (i.e. for $k = 0, \dots, 200$). Assume the human operator is aiming to minimise a quadratic cost functional (3.2), $i = 1$, with

$$Q_1 = \begin{bmatrix} 15 & 0 \\ 0 & 0.1 \end{bmatrix}, \quad R_{11} = 0.5 \quad \text{and} \quad R_{12} = 0,$$

whereas the robot is aiming to minimise (3.2), $i = 2$, with

$$Q_2 = \begin{bmatrix} 25 & 0 \\ 0 & 0.1 \end{bmatrix}, \quad R_{21} = 0 \quad \text{and} \quad R_{22} = 0.1.$$

This constitutes a two-player LQ discrete-time dynamic game of the form (3.1), (3.2), $i = 1, 2$. In line with the definition of Problem 5.2.1, each of the two players only has knowledge of their own cost parameters, but not of the cost parameters of the other player, i.e. the human operator only knows Q_1, R_{11}, and the robot only knows Q_2, R_{22}. Moreover, the system matrices A, B_1 and B_2 are considered unknown for the purpose of control design for both players. In contrast to Section 5.1.3, the human operator's control action is not fixed, but both human and robot iteratively update their strategies. To this end, the players take turns to collect data. While the robot's data collection experiments last for $n+m_2 = 3$ steps, i.e. 0.3 s, the human collects data for 5 steps, i.e. 0.5 s. During their data collection phases, each player gives exploring force inputs sampled randomly from a uniform distribution in the interval (-1 N, 1 N) to excite the system while the other player sticks to their current strategy. The time histories of the states and inputs (for both Algorithm 1 and Algorithm 2) are shown in Figure 5.5 and Figure 5.6, respectively. The chosen initial conditions are the stabilising feedback gains $K_1^{(0)} = \begin{bmatrix} 0 & 0 \end{bmatrix}$, $K_2^{(0)} = \begin{bmatrix} -0.97 & -3.75 \end{bmatrix}$, and $x_0 = \begin{bmatrix} -0.3 & 0 \end{bmatrix}^\top$. As above, the dotted grey vertical lines indicate the start of each new experiment for data collection and the solid grey vertical lines highlight the time instance at which the algorithms have converged with tolerance $\bar{\varepsilon} = 10^{-5}$. Both Algorithm 1 and Algorithm 2 converge to

$$K_1^\star = \begin{bmatrix} -1.03 & -0.51 \end{bmatrix}, \quad K_2^\star = \begin{bmatrix} -13.18 & -12.39 \end{bmatrix},$$

which satisfy (3.4), (3.5), $i = 1, 2$, and (3.6) with

$$P_1^\star = \begin{bmatrix} 106.96 & 29.41 \\ 29.41 & 13.80 \end{bmatrix}, \; P_2^\star = \begin{bmatrix} 224.10 & 88.55 \\ 88.55 & 78.42 \end{bmatrix},$$

where all values are rounded to two decimal places. For the remainder of the time horizon ($T_f = 200$, corresponding to 20 s) the players follow their determined equilibrium strategies, i.e. $u_1 = K_1^\star x$ and $u_2 = K_2^\star x$. Both Algorithms converge to the specified tolerance in approximately 7 s, and despite the iterative probing and strategy updates the considered reaching task is completed successfully. At

$k = 100$ (10 s) the target position changes from $x_t = 0$ to $x_t = -0.3$, initiating a second arm reaching movement back to the initial condition. The time histories in Figures 5.5 and 5.6 show how the human and the robot collaborate to achieve this second reaching movement while following their feedback Nash equilibrium strategies.

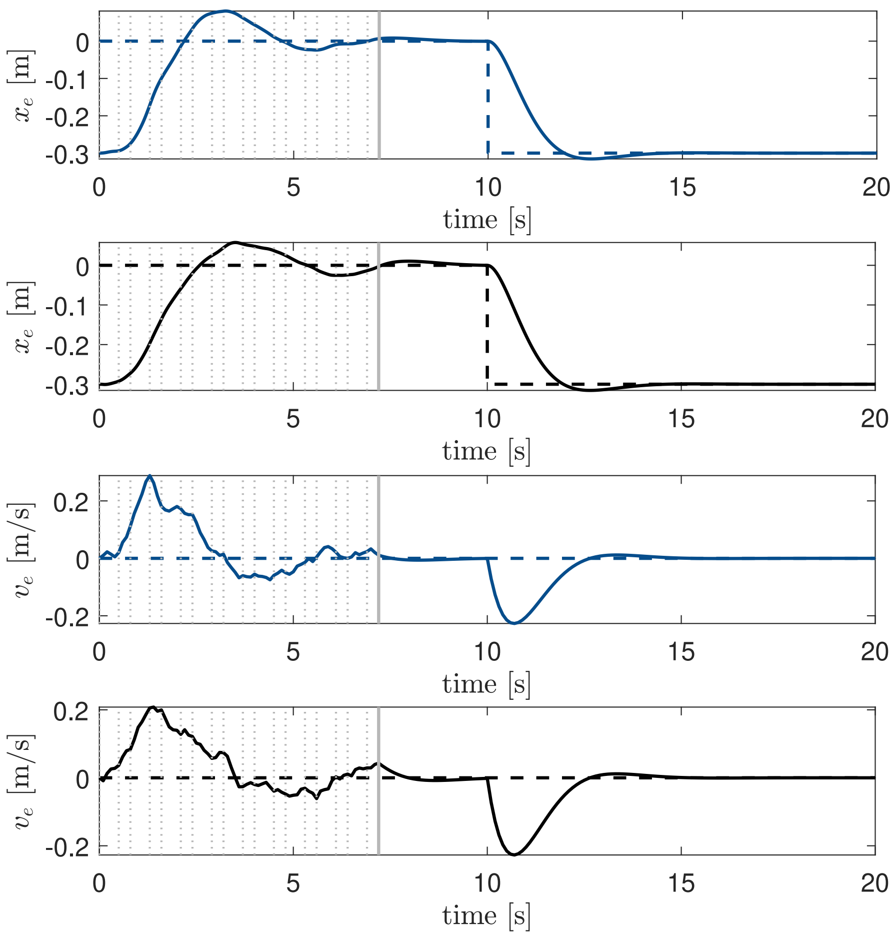


Figure 5.5: Time histories of the states using Algorithm 1 (dark blue) and Algorithm 2 (black). The dashed blue and black lines indicate the target values. The dotted grey lines indicate the start of each new experiment, whereas the solid grey line indicates the end of the scheduled experiments.

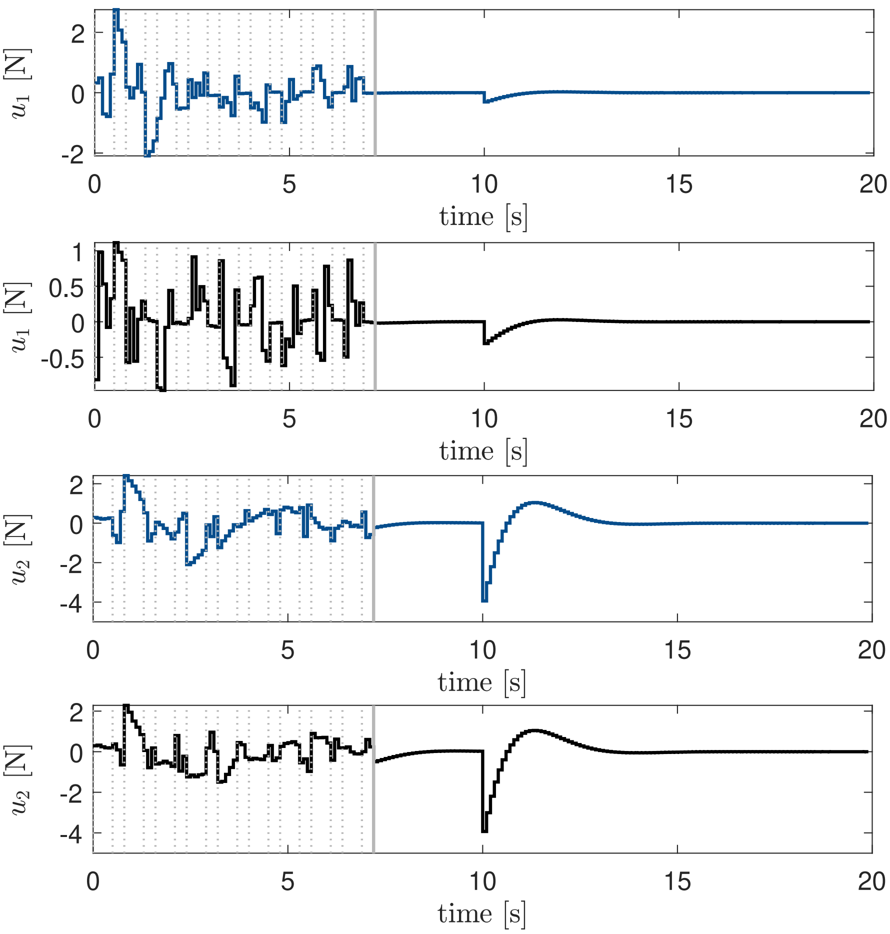


Figure 5.6: Time histories of the inputs using Algorithm 1 (dark blue) and Algorithm 2 (black). The dotted grey lines indicate the start of each new experiment, whereas the solid grey line indicates the end of the scheduled experiments.

5.3 Conclusion

Considering a class of LQ discrete-time dynamic games with asymmetric information structure, in which one player lacks information regarding the opponents' objectives and may not know the system dynamics, it is shown that data-driven results introduced in Chapter 4 can be leveraged to overcome the lack of information and determine a feedback Nash equilibrium solution of the game. The efficacy of the approach is demonstrated via simulations on a system involving human-robot interaction.

Considering instead the case in which all players lack information regarding the opponents' objectives and may not know the system dynamics, data-driven versions of the asynchronous algorithms presented in Section 3.4 are introduced. The algorithms allow the players to iteratively converge to a feedback Nash equilibrium of the game despite the incomplete information available to them by scheduling experiments. The approach is distributed in the sense that the players only require measurements of the state and their own inputs to update their strategies. The results are demonstrated and discussed via two illustrative numerical examples and a practical example involving human-robot interaction.

Chapter 6

Conclusion

Considering single-player and multi-player dynamic decision-making problems with complete and incomplete information, novel game theoretic and data-driven solution methods are proposed in this thesis. The main contributions are summarised as follows.

In Chapter 3, feedback Nash equilibrium solutions to infinite-horizon LQ discrete-time dynamic games with complete information are studied. Despite being of relevance in a variety of engineering and economics applications, this class of games has so far received less attention in the literature than the well studied continuous-time counterpart. The results presented in Chapter 3 contribute to filling this gap. First, the conditions characterising feedback Nash equilibria are recalled and the challenges associated with computing solutions for this class of games are highlighted. Namely, feedback Nash equilibria are characterised by the stabilising solutions of a set of coupled algebraic matrix equations, which in contrast to their continuous-time counterpart are not quadratic in the decision variables and involve additional coupling terms. For games involving scalar dynamics, graphical representations of the coupled algebraic equations are introduced. These representations not only allow to visualise the solutions and their properties, but also to derive conditions based on the system and cost parameters for the number of feedback Nash equilibrium solutions the game admits, as well as certain properties. Motivated by the challenges associated with computing feedback Nash equilibrium solutions, a notion of approximate Nash equilibrium is introduced (for games involving general linear, not necessarily scalar, dynamics), the $\epsilon_{\alpha,\beta}$-Nash equilibrium. The proposed notion of equilibrium allows to incorporate a guaranteed rate of convergence of the resulting closed-loop system. The degree of approximation and the computation of equilibria are discussed. The results are illustrated via a macroeconomic policy design example. Finally, an alternative approach to tackling the challenges associated with computing feedback Nash equilibria is presented. Four iterative methods to compute a Nash equilibrium solution of the game are proposed. The algorithms are based on the iterative solution of simpler, uncoupled matrix equations, namely Lyapunov or Riccati equations, for each player at each update iteration.

Conditions for local convergence to a Nash equilibrium are provided. The results of Chapter 3 address the first sub-objective listed in Section 1.1.

In Chapter 4, single-player decision-making problems with incomplete information are considered. Building on the direct data-driven control framework first introduced in [95] for LTI systems, novel data-driven methods are presented to overcome unknown system dynamics for the more general class of LTV systems, as well as to overcome unknown cost parameters in the context of LQR. First, a direct data-driven representation of state-feedback controllers and the corresponding closed-loop system is introduced for unknown LTV systems and employed to design controllers with trajectory boundedness, performance and robustness guarantees via purely data-dependent convex programmes. Challenges arising for control tasks involving large time horizons are addressed. While the results are applicable to LTV systems with arbitrary time-variation, it is shown that knowledge about the nature of the time-variation can be exploited for periodically time-varying systems to design infinite-horizon controllers based on finite data. The practical relevance of the results is motivated via case-studies involving control of a power system and locomotion control of a snake-like robot. Second, a direct data-driven representation of cost functions in the context of LQ optimal control is introduced. In contrast to inverse optimal control methods, the presented results do not rely on expert data, instead they utilise finite data sequences of the state and performance output response to non-optimal exploring inputs. Together with the system representation from [95] the proposed cost representation enables the solution of LQR problems in which both the system dynamics and the cost matrices are unknown via purely data-dependent convex programmes. While the results are presented for LTI systems, it is remarked how they can be extended to the LTV case by utilising similar arguments as in the first part of Chapter 4. By addressing single-player dynamic decision problems in the context of incomplete system and cost information, the results of Chapter 4 tackle the second sub-objective listed in Section 1.1.

In Chapter 5, multi-player dynamic decision problems with incomplete information are considered. By combining game theoretic methods from Chapter 3 and data-driven methods form Chapter 4, approaches to determine a feedback Nash equilibrium solution of an LQ discrete-time dynamic game, if one or more players lack knowledge of the performance criteria of other players and may not know the system dynamics, are introduced. First, it is shown that the cost representation results from Chapter 4 are relevant for a class of games with asymmetric information structure, in the sense that one player has incomplete system and cost information while all other players have full information. Next, the case in which all players have incomplete system and cost information is considered, and it is shown that similar arguments can be combined with the iterative methods proposed in Chapter 3 to develop data-driven algorithms, which allow the players to iteratively converge to a Nash equilibrium via scheduled experiments. Both results are demonstrated via a human-robot interaction example. This addresses the third sub-objective listed in Section 1.1.

Altogether, the ideas and applications presented in this thesis not only address interesting open problems in the fields of dynamic game theory and direct data-driven control, but also highlight the potential of combining game theoretic and direct data-driven methods to solve multi-player dynamic decision-making problems in the face of incomplete information.

In the following, interesting directions for future research on these topics are indicated.

Infinite-horizon dynamic games: The dynamic games results in this thesis focus on feedback Nash equilibrium solutions of the game. It would be interesting to conduct a similar analysis as in Chapter 3 for other solution concepts, for example feedback Stackelberg equilibrium solutions, which are a natural choice of solution to the game if there is a leader-follower hierarchy between the players. Initial preliminary results regarding the study of scalar two-player games via a graphical plane curve representation, as well as iterative Stackelberg equilibrium finding methods are promising but require further analysis.

Iterative methods: The iterative methods proposed in Sections 3.4 and 5.2 are provided with local convergence guarantees. Naturally, if multiple feedback Nash equilibrium solutions exist for the considered game, and Section 3.2 highlights how this can be the case even in the seemingly simple scalar two-player setting, then it is inherent that the convergence properties of any algorithm to a specific equilibrium can only be local. However, it would be interesting to study the "update dynamics" of the algorithms in more detail with the aim of better understanding the regions of attraction of different equilibria and hence to devise strategic ways to initialise the algorithms without prior knowledge of the solutions.

Direct data-driven control methods: The analysis of LTV systems can be seen as a step towards data-driven methods for nonlinear systems. Several approaches to extend the data-driven framework from [95] to classes of nonlinear systems have been proposed [138, 139, 140, 141, 142, 143, 144]. Most works in this contexts focus on stabilisation and only few on optimal control [139]. Even in the model-based setting, obtaining closed-form solutions to nonlinear optimal control problems can be challenging or even impossible. An interesting line of work would hence be to combine the data-driven system representation with the notion of "algebraic P solutions" [217], which allow the systematic construction of approximate solutions. Since the notion has also been extended to differential games, this might lay the path to new data-driven solutions for dynamic games.

Games for data-driven methods: The results in Chapter 5 highlight the potential of combining game theoretic and direct data-driven methods. Namely, it is demonstrated that introducing data-driven methods in the context of dynamic game problems makes it possible to overcome incomplete information. An interesting direction for future work includes exploring ways in which game theoretic tools can facilitate the solution of problems arising in data-driven control and analysis. Data science comprises many topics which involve multiple potentially conflicting objectives or the interaction of different decision makers. Hence, employing game theoretic methods is a promising approach to tackle challenges in this context. Examples of existing work in this direction include game theoretic approaches in the context of data collection (e.g. trajectory planning and control of autonomous robotic systems equipped with sensors to monitor or map an area [218], or design and control of crowdsensing platforms [219, 220]), distributed processing of “big data” [221], data quality control [222] and data trading [223]. However, this is an evolving field and many open problems remain [222].

Bibliography

[1] B. Brehmer, "Dynamic decision making: Human control of complex systems," *Acta Psychologica*, vol. 81, no. 3, pp. 211–241, 1992.

[2] S. Tadokoro, "Disaster response robot," in *Encyclopedia of Systems and Control*, J. Baillieul and T. Samad, Eds. London: Springer London, 2020, pp. 1–6.

[3] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "Snake robots: modelling, mechatronics, and control," in *Advances in Industrial Control*. Springer, 2013.

[4] P. Järventausta, S. Repo, A. Rautiainen, and J. Partanen, "Smart grid power system control in distributed generation environment," *Annual Reviews in Control*, vol. 34, no. 2, pp. 277–286, 2010.

[5] I. Colak, E. Kabalci, G. Fulli, and S. Lazarou, "A survey on the contributions of power electronics to smart grid systems," *Renewable and Sustainable Energy Reviews*, vol. 47, pp. 562–579, 2015.

[6] J. Chen, A. J. Gallo, S. Yan, T. Parisini, and S. Y. R. Hui, "Cyber-attack detection and countermeasure for distributed electric springs for smart grid applications," *IEEE Access*, vol. 10, pp. 13 182–13 192, 2022.

[7] H. Hjalmarsson, "From experiment design to closed-loop control," *Automatica*, vol. 41, no. 3, pp. 393–438, 2005.

[8] Z.-S. Hou and Z. Wang, "From model-based control to data-driven control: Survey, classification and perspective," *Information Sciences*, vol. 235, pp. 3–35, 2013.

[9] T. Başar and G. J. Olsder, *Dynamic Noncooperative Game Theory*, 2nd ed., ser. Classics in Applied Mathematics. Society for Industrial and Applied Mathematics, 1998, vol. 23.

[10] B. Van Aarle, J. C. Engwerda, J. E. Plasmans, and A. Weeren, "Macroeconomic policy interaction under EMU: A dynamic game approach," *Open Economies Review*, vol. 12, p. 29–60, 2001.

[11] R. Isaacs, *Differential Games: A Mathematical Theory with Application to Warfare and Pursuit, Control and Optimization.* John Wiley and Sons, New York, 1965.

[12] S. Jørgensen, G. Martín-Herrán, and G. Zaccour, "Dynamic games in the economics and management of pollution," *Environmental Modeling & Assessment*, vol. 15, pp. 433–467, 2010.

[13] I. Kordonis, A.-R. Lagos, and G. P. Papavassilopoulos, "Dynamic games of social distancing during an epidemic: Analysis of asymmetric solutions," *Dynamic Games and Applications*, vol. 12, pp. 214–236, 2022.

[14] Y. C. Ho, "Differential games, dynamic optimization, and generalized control theory," *Journal of Optimization Theory and Applications*, vol. 6, no. 3, pp. 179–209, Sep. 1970.

[15] T. Başar and P. Bernhard, *H-infinity optimal control and related minimax design problems: a dynamic game approach.* Springer Science & Business Media, 2008.

[16] D. Limebeer, B. Anderson, and B. Hendel, "A Nash game approach to mixed H_2/H_∞ control," *IEEE Transactions on Automatic Control*, vol. 39, no. 1, pp. 69–82, 1994.

[17] S. R. Etesami and T. Başar, "Dynamic games in cyber-physical security: An overview," *Dynamic Games and Applications*, vol. 9, pp. 884–913, 2019.

[18] T. Nagata and H. Sasaki, "A multi-agent approach to power system restoration," *IEEE Transactions on Power Systems*, vol. 17, no. 2, pp. 457–462, 2002.

[19] V. P. Singh, N. Kishor, and P. Samuel, "Distributed multi-agent system-based load frequency control for multi-area power system in smart grid," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 5151–5160, 2017.

[20] Y. Li, G. Carboni, F. Gonzalez, D. Campolo, and E. Burdet, "Differential game theory for versatile physical human–robot interaction," *Nature Machine Intelligence*, vol. 1, no. 1, pp. 36–43, 2019.

[21] S. Musić and S. Hirche, "Haptic shared control for human-robot collaboration: A game-theoretical approach," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 10 216–10 222, 2020.

[22] B. Varga, J. Inga, M. Lemmer, and S. Hohmann, "Ordinal potential differential games to model human-machine interaction in vehicle-manipulators," in *IEEE Conference on Control Technology and Applications*, 2021, pp. 728–734.

[23] T. Mylvaganam and M. Sassano, "Autonomous collision avoidance for wheeled mobile robots using a differential game approach," *European Journal of Control*, vol. 40, pp. 53–61, 2018.

[24] M. Wrzos-Kaminska, T. Mylvaganam, K. Y. Pettersen, and J. Tommy Gravdahl, "Collision avoidance using mixed H_2/H_∞ control for an articulated intervention-AUV," in *European Control Conference*, 2020, pp. 881–888.

[25] T. Mylvaganam, M. Sassano, and A. Astolfi, "A differential game approach to multi-agent collision avoidance," *IEEE Transactions on Automatic Control*, vol. 62, pp. 4229–4235, 2017.

[26] D. Cappello, S. Garcin, Z. Mao, M. Sassano, A. Paranjape, and T. Mylvaganam, "A hybrid controller for multi-agent collision avoidance via a differential game formulation," *IEEE Transactions on Control Systems Technology*, vol. 29, pp. 1750–1757, 2021.

[27] D. Gu, "A differential game approach to formation control," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 1, pp. 85–93, 2008.

[28] Q. Jiao, H. Modares, S. Xu, F. L. Lewis, and K. G. Vamvoudakis, "Multi-agent zero-sum differential graphical games for disturbance rejection in distributed control," *Automatica*, vol. 69, pp. 24–34, 2016.

[29] D. Cappello and T. Mylvaganam, "Distributed differential games for control of multi-agent systems," *IEEE Transactions on Control of Network Systems*, 2021.

[30] J. Engwerda, *LQ Dynamic Optimization and Differential Games.* John Wiley & Sons, 2005.

[31] A. Monti, B. Nortmann, T. Mylvaganam, and M. Sassano, "Feedback and open-loop Nash equilibria for LQ infinite-horizon discrete-time dynamic games," *SIAM Journal on Control and Optimization*, vol. 62, no. 3, pp. 1417–1436, 2024.

[32] M. L. Scarpa, B. Nortmann, K. Y. Pettersen, and T. Mylvaganam, "Data-driven control of planar snake robot locomotion," in *IEEE Conference on Decision and Control*, 2022, pp. 1823–1828.

[33] D. Ross, "Game theory," in *The Stanford Encyclopedia of Philosophy*, E. N. Zalta, Ed., 2021. [Online]. Available: https://plato.stanford.edu/archives/fall2021/entries/game-theory/

[34] J. von Neumann and O. Morgenstern, *Theory of games and economic behavior.* Princeton University Press, 1944.

[35] A. W. Starr and Y. C. Ho, "Nonzero-sum differential games," *Journal of Optimization Theory and Applications*, vol. 3, pp. 184–206, 1969.

[36] J. F. Nash, Jr., "Equilibrium points in n-person games," *Proceedings of the National Academy of Sciences*, vol. 36, no. 1, pp. 48–49, 1950.

[37] J. Nash, "Non-cooperative games," *Annals of Mathematics*, vol. 54, no. 2, pp. 286–295, 1951.

[38] V. Pareto, *Cours d'économie politique. Vol. 1 and 2.* Librairie de l'Université Lausanne, 1896.

[39] H. von Stackelberg, *Market Structure and Equilibrium.* Springer Berlin Heidelberg, 2011, translated from German by Damien Bazin, Lynn Urch and Rowland Hill.

[40] A. Haurie, "A historical perspective on cooperative differential games," in *Advances in Dynamic Games and Applications*, E. Altman and O. Pourtallier, Eds. Boston, MA: Birkhäuser Boston, 2001, pp. 19–29.

[41] W. Thomson, "Cooperative models of bargaining," in *Handbook of Game Theory with Economic Applications.* Elsevier, 1994, vol. 2, pp. 1237–1284.

[42] G. Neumann and S. Schuster, "Continuous model for the rock–scissors–paper game between bacteriocin producing bacteria," *Journal of Mathematical Biology*, vol. 54, no. 6, pp. 815–846, 2007.

[43] N. V. Long, "Dynamic games in the economics of natural resources: A survey," *Dynamic Games and Applications*, vol. 1, pp. 115–148, 2011.

[44] S. Jørgensen, M. Quincampoix, and T. Vincent, *Advances in Dynamic Game Theory: Numerical Methods, Algorithms, and Applications to Ecology and Economics*, ser. Annals of the International Society of Dynamic Games. Birkhäuser Boston, 2007.

[45] E. Dockner, S. Jørgensen, N. Long, and G. Sorger, *Differential Games in Economics and Management Science.* Cambridge University Press, 2000.

[46] M. Nadini, L. Zino, A. Rizzo, and M. Porfiri, "A multi-agent model to study epidemic spreading and vaccination strategies in an urban-like environment," *Applied Network Science*, vol. 5, pp. 68–97, 2020.

[47] M. Flad, "Differential-game-based driver assistance system for fuel-optimal driving," in *Frontiers of Dynamic Games: Game Theory and Management, St. Petersburg, 2018*, L. A. Petrosyan, V. V. Mazalov, and N. A. Zenkevich, Eds. Springer International Publishing, 2019, pp. 13–36.

[48] E. Calvo, "Dynamic models of international environmental agreements: A differential game approach," *International Review of Environmental and Resource Economics*, vol. 6, no. 4, pp. 289–339, 2013.

[49] T. Mylvaganam and A. Astolfi, "A Nash game approach to mixed H_2/H_∞ control for input-affine nonlinear systems," *IFAC-PapersOnLine*, vol. 49, no. 18, pp. 1024–1029, 2016.

[50] R. Bellman, *Dynamic Programming.* Princeton University Press, 1957.

[51] L. S. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko, *The Mathematical Theory of Optimal Processes.* Interscience Publishers, 1962.

[52] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal control.* John Wiley & Sons, 2012.

[53] T. Başar, "A counterexample in linear-quadratic games: Existence of nonlinear Nash solutions," *Journal of Optimization Theory and Applications*, vol. 14, no. 4, pp. 425–430, 1974.

[54] ——, "On the uniqueness of the Nash solution in linear-quadratic differential games," *International Journal of Game Theory*, vol. 5, no. 2-3, pp. 65–90, 1976.

[55] J. C. Engwerda and Salmah, "Necessary and sufficient conditions for feedback Nash equilibria for the affine-quadratic differential game," *Journal of Optimization Theory and Applications*, vol. 157, no. 2, pp. 552–563, 2012.

[56] C. Possieri and M. Sassano, "An algebraic geometry approach for the computation of all linear feedback Nash equilibria in LQ differential games," in *IEEE Conference on Decision and Control*, 2015, pp. 5197–5202.

[57] G. P. Papavassilopoulos, J. V. Medanic, and J. B. Cruz, "On the existence of Nash strategies and solutions to coupled Riccati equations in linear-quadratic games," *Journal of Optimization Theory and Applications*, vol. 28, no. 1, pp. 49–76, May 1979.

[58] J. Engwerda, "Properties of feedback Nash equilibria in scalar LQ differential games," *Automatica*, vol. 69, pp. 364–374, 2016.

[59] M. Pachter and K. D. Pham, "Discrete-time linear-quadratic dynamic games," *Journal of Optimization Theory and Applications*, vol. 146, no. 1, pp. 151–179, 2010.

[60] M. Jimenez and A. Poznyak, "ϵ-Equilibrium in LQ differential games with bounded uncertain disturbances: robustness of standard strategies and new strategies with adaptation," *International Journal of Control*, vol. 79, no. 7, pp. 786–797, 2006.

[61] T. Mylvaganam, M. Sassano, and A. Astolfi, "Constructive ϵ-Nash equilibria for nonzero-sum differential games," *IEEE Transactions on Automatic Control*, vol. 60, no. 4, pp. 950–965, 2015.

[62] D. Cappello and T. Mylvaganam, "Approximate Nash equilibrium solutions of linear quadratic differential games," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 6685–6690, 2020.

[63] K. G. Vamvoudakis and F. L. Lewis, "Multi-player non-zero-sum games: Online adaptive learning solution of coupled Hamilton–Jacobi equations," *Automatica*, vol. 47, no. 8, pp. 1556–1569, 2011.

[64] R. Song, F. L. Lewis, and Q. Wei, "Off-policy integral reinforcement learning method to solve nonlinear continuous-time multiplayer nonzero-sum games," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 3, pp. 704–713, 2017.

[65] J. B. Cruz and C. I. Chen, "Series Nash solution of two-person, nonzero-sum, linear-quadratic differential games," *Journal of Optimization Theory and Applications*, vol. 7, no. 4, pp. 240–257, 1971.

[66] T.-Y. Li and Z. Gajic, "Lyapunov iterations for solving coupled algebraic Riccati equations of Nash differential games and algebraic Riccati equations of zero-sum games," in *New Trends in Dynamic Games and Applications.* Birkhäuser Boston, 1995, pp. 333–351.

[67] G. Freiling, G. Jank, and H. Abou-Kandil, "On global existence of solutions to coupled matrix Riccati equations in closed-loop Nash games," *IEEE Transactions on Automatic Control*, vol. 41, no. 2, pp. 264–269, 1996.

[68] J. Engwerda, "Algorithms for computing Nash equilibria in deterministic LQ games," *Computational Management Science*, vol. 4, no. 2, pp. 113–140, 2007.

[69] A. Odekunle, W. Gao, M. Davari, and Z.-P. Jiang, "Reinforcement learning and non-zero-sum game output regulation for multi-player linear uncertain systems," *Automatica*, vol. 112, p. 108672, 2020.

[70] D. A. Behrens and R. Neck, "Optgame: An algorithm approximating solutions for multi-player difference games," in *Advances and Innovations in Systems, Computing Sciences and Software Engineering*, K. Elleithy, Ed. Springer Netherlands, 2007, pp. 93–98.

[71] H. Zhang, H. Jiang, C. Luo, and G. Xiao, "Discrete-time nonzero-sum games for multiplayer using policy-iteration-based adaptive dynamic programming algorithms," *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3331–3340, 2017.

[72] J. Li, Z. Xiao, and P. Li, "Discrete-time multi-player games based on off-policy Q-learning," *IEEE Access*, vol. 7, pp. 134 647–134 659, 2019.

[73] Y. Yang, S. Zhang, J. Dong, and Y. Yin, "Data-driven nonzero-sum game for discrete-time systems using off-policy reinforcement learning," *IEEE Access*, vol. 8, pp. 14 074–14 088, 2020.

[74] R. Song, Q. Wei, H. Zhang, and F. L. Lewis, "Discrete-time non-zero-sum games with completely unknown dynamics," *IEEE Transactions on Cybernetics*, vol. 51, no. 6, pp. 2929–2943, 2021.

[75] D. Vrabie and F. Lewis, "Integral reinforcement learning for online computation of feedback Nash strategies of nonzero-sum differential games," in *IEEE Conference on Decision and Control*, 2010, pp. 3066–3071.

[76] H. Jiang, H. Zhang, K. Zhang, and X. Cui, "Data-driven adaptive dynamic programming schemes for non-zero-sum games of unknown discrete-time nonlinear systems," *Neurocomputing*, vol. 275, pp. 649–658, 2018.

[77] H. Orojloo and M. A. Azgomi, "A game-theoretic approach to model and quantify the security of cyber-physical systems," *Computers in Industry*, vol. 88, pp. 44–57, 2017.

[78] M. I. Abouheaf, F. L. Lewis, K. G. Vamvoudakis, S. Haesaert, and R. Babuska, "Multi-agent discrete-time graphical games and reinforcement learning solutions," *Automatica*, vol. 50, no. 12, pp. 3038–3053, 2014.

[79] R. Kamalapurkar, J. R. Klotz, P. Walters, and W. E. Dixon, "Model-based reinforcement learning in differential graphical games," *IEEE Transactions on Control of Network Systems*, vol. 5, no. 1, pp. 423–433, 2018.

[80] K. G. Vamvoudakis, "Q-learning for continuous-time graphical games on large networks with completely unknown linear system dynamics," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 16, pp. 2900–2920, 2016.

[81] T. L. Molloy, J. I. Charaja, S. Hohmann, and T. Perez, "Inverse noncooperative dynamic games," in *Inverse Optimal Control and Inverse Noncooperative Dynamic Game Theory.* Springer International Publishing, 2022, pp. 143–187.

[82] K. J. Åström, *Introduction to stochastic control theory.* Academic Press, 1970.

[83] R. Buckdahn, P. Cardaliaguet, and C. Rainer, "Nash equilibrium payoffs for nonzero-sum stochastic differential games," *SIAM Journal on Control and Optimization*, vol. 43, no. 2, pp. 624–642, 2004.

[84] T. Başar, "Decentralized multicriteria optimization of linear stochastic systems," *IEEE Transactions on Automatic Control*, vol. 23, no. 2, pp. 233–243, 1978.

[85] T. Başar and S. Li, "Distributed computation of Nash equilibria in linear-quadratic stochastic differential games," *SIAM Journal on Control and Optimization*, vol. 27, no. 3, pp. 563–578, 1989.

[86] B. Yongacoglu, G. Arslan, and S. Yüksel, "Decentralized learning for optimality in stochastic dynamic teams and games with local control and global state information," *IEEE Transactions on Automatic Control*, vol. 67, no. 10, pp. 5230–5245, 2022.

[87] Y. Jiang, W. Cui, B. Zhang, and J. Cortes, "Equilibria of fully decentralized learning in networked systems," in *Learning for Dynamics and Control Conference*, N. Matni, M. Morari, and G. J. Pappas, Eds., vol. 211, 2023, pp. 333–345.

[88] D. D. Šiljak, *Decentralized Control of Complex Systems.* Academic Press, 1991.

[89] L. Bakule, "Decentralized control: An overview," *Annual Reviews in Control*, vol. 32, no. 1, pp. 87–98, 2008.

[90] Y.-C. Ho, “Team decision theory and information structures,” *Proceedings of the IEEE*, vol. 68, no. 6, pp. 644–654, 1980.

[91] M. Verhaegen and V. Verdult, *Filtering and System Identification: A Least Squares Approach.* Cambridge University Press, 2007.

[92] A. Chiuso and G. Pillonetto, “System identification: A machine learning perspective,” *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, no. 1, pp. 281–304, 2019.

[93] K. Åström and B. Wittenmark, *Adaptive Control*, 2nd ed. Dover Publications, 2008.

[94] K.-S. Fu, “Learning control systems–review and outlook,” *IEEE Transactions on Automatic Control*, vol. 15, no. 2, pp. 210–221, 1970.

[95] C. De Persis and P. Tesi, “Formulas for data-driven control: Stabilization, optimality, and robustness,” *IEEE Transactions on Automatic Control*, vol. 65, no. 3, pp. 909–924, 2020.

[96] J. G. Ziegler and N. B. Nichols, “Optimum settings for automatic controllers,” *Transactions of the ASME*, vol. 64, pp. 759–768, 1942.

[97] M. G. Safonov and T.-C. Tsao, “The unfalsified control concept: A direct path from experiment to controller,” in *Feedback Control, Nonlinear Systems, and Complexity. Lecture Notes in Control and Information Sciences*, B. Francis and A. Tannenbaum, Eds., 1995, vol. 202, pp. 196–214.

[98] H. Hjalmarsson, M. Gevers, S. Gunnarsson, and O. Lequin, “Iterative feedback tuning: theory and applications,” *IEEE Control Systems Magazine*, vol. 18, no. 4, pp. 26–41, 1998.

[99] M. Campi, A. Lecchini, and S. Savaresi, “Virtual reference feedback tuning: a direct method for the design of feedback controllers,” *Automatica*, vol. 38, no. 8, pp. 1337–1346, 2002.

[100] Z. Hou and S. Jin, “Data-driven model-free adaptive control for a class of MIMO nonlinear discrete-time systems,” *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2173–2188, 2011.

[101] Z. Hou, H. Gao, and F. L. Lewis, “Data-driven control and learning systems,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 5, pp. 4070–4075, 2017.

[102] H. J. V. Waarde, J. Eising, H. L. Trentelman, and M. K. Camlibel, “Data informativity: a new perspective on data-driven analysis and control,” *IEEE Transactions on Automatic Control*, vol. 65, no. 11, pp. 4753–4768, 2020.

[103] P. J. Werbos, “Neural networks for control and system identification,” in *IEEE Conference on Decision and Control*, 1989, pp. 260–265.

[104] B. Recht, "A tour of reinforcement learning: The view from continuous control," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 2, no. 1, pp. 253–279, 2019.

[105] S. Meyn, *Control Systems and Reinforcement Learning.* Cambridge University Press, May 2022.

[106] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. D. Moor, "A note on persistency of excitation," *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, 2005.

[107] J. C. Willems, "From time series to linear system—part I. finite dimensional linear time invariant systems," *Automatica*, vol. 22, no. 5, pp. 561–580, 1986.

[108] ——, "From time series to linear system—part II. exact modelling," *Automatica*, vol. 22, no. 6, pp. 675–694, 1986.

[109] ——, "From time series to linear system—part III," *Automatica*, vol. 23, no. 1, pp. 87–115, 1987.

[110] I. Markovsky and P. Rapisarda, "On the linear quadratic data-driven control," in *European Control Conference*, 2007, pp. 5313–5318.

[111] ——, "Data-driven simulation and control," *International Journal of Control*, vol. 81, no. 12, pp. 1946–1959, 2008.

[112] U. S. Park and M. Ikeda, "Stability analysis and control design of LTI discrete-time systems by the direct use of time series data," *Automatica*, vol. 45, no. 5, pp. 1265–1271, 2009.

[113] J. Coulson, J. Lygeros, and F. Dörfler, "Data-enabled predictive control: In the shallows of the DeePC," in *European Control Conference*, 2019, pp. 307–312.

[114] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2021.

[115] S. Boyd and L. Vandenberghe, *Convex Optimization.* Cambridge University Press, 2004.

[116] H. J. van Waarde, J. Eising, M. K. Camlibel, and H. L. Trentelman, "A tutorial on the informativity framework for data-driven control," in *IEEE Conference on Decision and Control*, 2022, pp. 1085–1090.

[117] ——, "The informativity approach: To data-driven analysis and control," *IEEE Control Systems Magazine*, vol. 43, no. 6, pp. 32–66, 2023.

[118] I. Markovsky and F. Dörfler, "Behavioral systems theory in data-driven analysis, signal processing, and control," *Annual Reviews in Control*, vol. 52, pp. 42–64, 2021.

[119] H. J. van Waarde, C. De Persis, M. K. Camlibel, and P. Tesi, "Willems' fundamental lemma for state-space systems and its extension to multiple datasets," *IEEE Control Systems Letters*, vol. 4, no. 3, pp. 602–607, 2020.

[120] Y. Yu, S. Talebi, H. J. van Waarde, U. Topcu, M. Mesbahi, and B. Acikmese, "On controllability and persistency of excitation in data-driven control: Extensions of Willems' fundamental lemma," in *IEEE Conference on Decision and Control*, 2021, pp. 6485–6490.

[121] J. Berberich, S. Wildhagen, M. Hertneck, and F. Allgöwer, "Data-driven analysis and control of continuous-time systems under aperiodic sampling," *IFAC-PapersOnLine*, vol. 54, no. 7, pp. 210–215, 2021.

[122] V. G. Lopez and M. A. Müller, "On a continuous-time version of Willems' lemma," in *IEEE Conference on Decision and Control*, 2022, pp. 2759–2764.

[123] P. Rapisarda, M. Çamlibel, and H. van Waarde, "A "fundamental lemma" for continuous-time systems, with applications to data-driven simulation," *Systems & Control Letters*, vol. 179, p. 105603, 2023.

[124] T. R. V. Steentjes, M. Lazar, and P. M. J. V. den Hof, "On data-driven control: Informativity of noisy input-output data with cross-covariance bounds," *IEEE Control Systems Letters*, vol. 6, pp. 2192–2197, 2022.

[125] J. Berberich, C. W. Scherer, and F. Allgöwer, "Combining prior knowledge and data for robust controller design," *IEEE Transactions on Automatic Control*, vol. 68, no. 8, pp. 4618–4633, 2023.

[126] H. van Waarde, J. Eising, M. Camlibel, and H. Trentelman, "A behavioral approach to data-driven control with noisy input-output data," *IEEE Transactions on Automatic Control*, pp. 1–14, 2023.

[127] C. Verhoek, R. Tóth, S. Haesaert, and A. Koch, "Fundamental lemma for data-driven analysis of linear parameter-varying systems," in *IEEE Conference on Decision and Control*, 2021, pp. 5040–5046.

[128] C. Verhoek, J. Berberich, S. Haesaert, F. Allgöwer, and R. Tóth, "Data-driven dissipativity analysis of linear parameter-varying systems," *IEEE Transactions on Automatic Control*, 2024.

[129] C. Verhoek, R. Tóth, and H. S. Abbas, "Direct data-driven state-feedback control of linear parameter-varying systems," *arXiv preprint:* 10.48550/arXiv.2211.17182, 2024.

[130] M. Rotulo, C. De Persis, and P. Tesi, "Online learning of data-driven controllers for unknown switched linear systems," *Automatica*, vol. 145, p. 110519, 2022.

[131] J. G. Rueda-Escobedo, E. Fridman, and J. Schiffer, "Data-driven control for linear discrete-time delay systems," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3321–3336, 2022.

[132] X. Wang, J. Berberich, J. Sun, G. Wang, F. Allgöwer, and J. Chen, "Model-based and data-driven control of event- and self-triggered discrete-time linear systems," *IEEE Transactions on Cybernetics*, vol. 53, no. 9, pp. 6066–6079, 2023.

[133] C. De Persis, R. Postoyan, and P. Tesi, "Event-triggered control from data," *IEEE Transactions on Automatic Control*, 2023.

[134] G. Pan, R. Ou, and T. Faulwasser, "On a stochastic fundamental lemma and its use for data-driven optimal control," *IEEE Transactions on Automatic Control*, vol. 68, no. 10, pp. 5922–5937, 2023.

[135] J. Jiao, H. J. van Waarde, H. L. Trentelman, M. K. Camlibel, and S. Hirche, "Data-driven output synchronization of heterogeneous leader-follower multi-agent systems," in *IEEE Conference on Decision and Control*. IEEE, 2021, pp. 466–471.

[136] A. Allibhoy and J. Cortes, "Data-based receding horizon control of linear network systems," *IEEE Control Systems Letters*, vol. 5, no. 4, pp. 1207–1212, 2021.

[137] G. Baggio, D. S. Bassett, and F. Pasqualetti, "Data-driven control of complex networks," *Nature Communications*, vol. 12, no. 1, 2021.

[138] A. Bisoffi, C. De Persis, and P. Tesi, "Data-based stabilization of unknown bilinear systems with guaranteed basin of attraction," *Systems & Control Letters*, vol. 145, p. 104788, 2020.

[139] Z. Yuan and J. Cortes, "Data-driven optimal control of bilinear systems," *IEEE Control Systems Letters*, vol. 6, pp. 2479–2484, 2022.

[140] M. Guo, C. De Persis, and P. Tesi, "Data-driven stabilization of nonlinear polynomial systems with noisy data," *IEEE Transactions on Automatic Control*, vol. 67, no. 8, pp. 4210–4217, 2022.

[141] M. Alsalti, V. G. Lopez, J. Berberich, F. Allgöwer, and M. A. Müller, "Data-based control of feedback linearizable systems," *IEEE Transactions on Automatic Control*, pp. 1–8, 2023.

[142] M. Guo, C. De Persis, and P. Tesi, "Data-driven stabilizer design and closed-loop analysis of general nonlinear systems via Taylor's expansion," *arXiv preprint:* 10.48550/arxiv.2209.01071, 2023.

[143] C. De Persis, M. Rotulo, and P. Tesi, "Learning controllers from data via approximate nonlinearity cancellation," *IEEE Transactions on Automatic Control*, vol. 68, no. 10, pp. 6082–6097, 2023.

[144] X. Dai, C. De Persis, N. Monshizadeh, and P. Tesi, "Data-driven control of nonlinear systems from input-output data," in *IEEE Conference on Decision and Control*, 2023, pp. 1613–1618.

[145] A. Luppi, C. De Persis, and P. Tesi, "On data-driven stabilization of systems with nonlinearities satisfying quadratic constraints," *Systems & Control Letters*, vol. 163, p. 105206, 2022.

[146] J. Berberich, A. Koch, C. W. Scherer, and F. Allgöwer, "Robust data-driven state-feedback design," in *American Control Conference*, 2020, pp. 1532–1538.

[147] H. J. van Waarde, M. K. Camlibel, J. Eising, and H. L. Trentelman, "Quadratic matrix inequalities with applications to data-based control," *SIAM Journal on Control and Optimization*, vol. 61, no. 4, pp. 2251–2281, 2023.

[148] H. J. van Waarde, M. K. Camlibel, and M. Mesbahi, "From noisy data to feedback controllers: non-conservative design via a matrix S-lemma," *IEEE Transactions on Automatic Control*, vol. 64, no. 1, pp. 162–175, 2022.

[149] H. J. van Waarde and M. K. Camlibel, "A matrix Finsler's lemma with applications to data-driven control," in *IEEE Conference on Decision and Control*, 2021, pp. 5777–5782.

[150] A. Bisoffi, C. De Persis, and P. Tesi, "Data-driven control via Petersen's lemma," *Automatica*, vol. 145, p. 110537, 2022.

[151] H. J. van Waarde, M. K. Camlibel, P. Rapisarda, and H. L. Trentelman, "Data-driven dissipativity analysis: Application of the matrix S-lemma," *IEEE Control Systems*, vol. 42, no. 3, pp. 140–149, 2022.

[152] A. Bisoffi, C. De Persis, and P. Tesi, "Trade-offs in learning controllers from noisy data," *Systems & Control Letters*, vol. 154, p. 104985, 2021.

[153] A. Bisoffi, C. De Persis, and P. Tesi, "Controller design for robust invariance from noisy data," *IEEE Transactions on Automatic Control*, vol. 68, no. 1, pp. 636–643, 2023.

[154] T. Martin and F. Allgöwer, "Dissipativity verification with guarantees for polynomial systems from noisy input-state data," *IEEE Control Systems Letters*, vol. 5, no. 4, pp. 1399–1404, 2021.

[155] R. Strässer, J. Berberich, and F. Allgöwer, "Robust data-driven control for nonlinear systems using the Koopman operator," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 2257–2262, 2023.

[156] C. De Persis and P. Tesi, "Low-complexity learning of linear quadratic regulators from noisy data," *Automatica*, vol. 128, p. 109548, 2021.

[157] F. Dörfler, P. Tesi, and C. De Persis, "On the role of regularization in direct data-driven LQR control," in *IEEE Conference on Decision and Control*, 2022.

[158] F. Dörfler, J. Coulson, and I. Markovsky, "Bridging direct and indirect data-driven control formulations via regularizations and relaxations," *IEEE Transactions on Automatic Control*, vol. 68, no. 2, pp. 883–897, 2023.

[159] F. Dörfler, P. Tesi, and C. De Persis, "On the certainty-equivalence approach to direct data-driven LQR design," *IEEE Transactions on Automatic Control*, pp. 1–8, 2023.

[160] A. Padoan, J. Coulson, H. J. van Waarde, J. Lygeros, and F. Dörfler, "Behavioral uncertainty quantification for data-driven control," in *IEEE 61st Conference on Decision and Control*, 2022.

[161] J. Coulson, H. van Waarde, and F. Dörfler, "Robust fundamental lemma for data-driven control," *arXiv preprint:* 10.48550/arxiv.2205.06636, 2022.

[162] J. Berberich, A. Iannelli, A. Padoan, J. Coulson, F. Dörfler, and F. Allgöwer, "A quantitative and constructive proof of Willems' fundamental lemma and its implications," in *American Control Conference*, 2023, pp. 4155–4160.

[163] J. Coulson, H. J. van Waarde, J. Lygeros, and F. Dörfler, "A quantitative notion of persistency of excitation and the robust fundamental lemma," *IEEE Control Systems Letters*, vol. 7, pp. 1243–1248, 2023.

[164] R. E. Kalman, "When is a linear control system optimal?" *Journal of Basic Engineering*, vol. 86, no. 1, pp. 51–60, 1964.

[165] F. Nori and R. Frezza, "Linear optimal control problems and quadratic cost functions estimation," in *IEEE Mediterranean Conference on Control and Automation*, 2004.

[166] M. C. Priess, R. Conway, J. Choi, J. M. Popovich, and C. Radcliffe, "Solutions to the inverse LQR problem with application to biological systems analysis," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 2, pp. 770–777, 2015.

[167] B. Berret, E. Chiovetto, F. Nori, and T. Pozzo, "Evidence for composite cost functions in arm movement planning: An inverse optimal control approach," *PLoS Computational Biology*, vol. 7, no. 10, p. e1002183, 2011.

[168] M. Menner, P. Worsnop, and M. N. Zeilinger, "Constrained inverse optimal control with application to a human manipulation task," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 2, pp. 826–834, 2021.

[169] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion—an inverse optimal control approach," *Autonomous Robots*, vol. 28, no. 3, pp. 369–383, 2009.

[170] I. A. Faruque, F. T. Muijres, K. M. Macfarlane, A. Kehlenbeck, and J. S. Humbert, "Identification of optimal feedback control rules from micro-quadrotor and insect flight trajectories," *Biological Cybernetics*, vol. 112, no. 3, pp. 165–179, 2018.

[171] N. Yokoyama, "Inference of aircraft intent via inverse optimal control including second-order optimality condition," *AIAA Journal of Guidance, Control, and Dynamics*, vol. 41, no. 2, pp. 349–359, 2018.

[172] E. Kamen, "Fundamentals of linear time-varying systems," in *The Control Systems Handbook*, 2nd ed., W. S. Levine, Ed. CRC Press, 2010, ch. 3.

[173] A. Haurie, J. B. Krawczyk, and G. Zaccour, *Games and dynamic games.* World Scientific Publishing Company, 2012, vol. 1.

[174] R. Bix, *Conics and Cubics: A Concrete Introduction to Algebraic Curves*, 2nd ed. Springer New York, 2006.

[175] C.-H. Lee, "Upper and lower matrix bounds of the solution for the discrete Lyapunov equation," *IEEE Transactions on Automatic Control*, vol. 41, no. 9, pp. 1338–1341, 1996.

[176] C. Sun and R. Dai, "Rank-constrained optimization and its applications," *Automatica*, vol. 82, pp. 128–136, 2017.

[177] M. Fazel, H. Hindi, and S. P. Boyd, "Log-det heuristic for matrix rank minimization with applications to Hankel and Euclidean distance matrices," in *American Control Conference*, vol. 3, 2003, pp. 2156–2162.

[178] P. Cull, M. Flahive, and R. Robson, *Difference Equations: From Rabbits to Chaos.* Springer-Verlag, 2005.

[179] H. Khalil, *Nonlinear Systems*, 2nd ed. Prentice Hall, 1996.

[180] G. Spelsberg-Korspeter, D. Hochlenert, E. Heffel, A. Wagner, P. Hagedorn, and R. Sampaio, "Construction of Lyapunov functions for the estimation of basins of attraction," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 34, p. 633–639, 2012.

[181] R. Genesio, M. Tartaglia, and A. Vicino, "On the estimation of asymptotic stability regions: State of the art and new proposals," *IEEE Transactions on Automatic Control*, vol. 30, no. 8, pp. 747–755, 1985.

[182] H. K. Khalil, *High-Gain Observers in Nonlinear Feedback Control.* SIAM, Jun. 2017.

[183] A. Stuart and A. R. Humphries, *Dynamical systems and numerical analysis.* Cambridge University Press, 1996.

[184] H.-D. Chiang and L. F. C. Alberto, *Stability Regions of Nonlinear Dynamical Systems: Theory, Estimation, and Applications.* Cambridge University Press, 2015.

[185] J. Kapinski and J. Deshmukh, *Discovering Forward Invariant Sets for Nonlinear Dynamical Systems.* Springer International Publishing, 2015, p. 259–264.

[186] O. G. Antonovskaya and V. I. Goryunov, "On a method of evaluation of attraction domain for fixed point of nonlinear point mapping of arbitrary dimension," *Russian Mathematics*, vol. 60, no. 12, p. 9–14, 2016.

[187] R. Bobiti and M. Lazar, "Automated-sampling-based stability verification and DOA estimation for nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 63, no. 11, pp. 3659–3674, 2018.

[188] A. Lederer and S. Hirche, "Local asymptotic stability analysis and region of attraction estimation with Gaussian processes," in *IEEE Conference on Decision and Control*, 2019, pp. 1766–1771.

[189] A. Barraud, "A numerical algorithm to solve $A^T XA$-$X= Q$," *IEEE Transactions on Automatic Control*, vol. 22, no. 5, pp. 883–885, 1977.

[190] D. P. Bertsekas, *Reinforcement learning and optimal control.* Belmont, Massachusetts: Athena Scientific, 2019.

[191] J. Fong, Y. Tan, V. Crocher, D. Oetomo, and I. Mareels, "Dual-loop iterative optimal control for the finite horizon LQR problem with unknown dynamics," *Systems & Control Letters*, vol. 111, pp. 49–57, 2018.

[192] S.-J. Liu, M. Krstic, and T. Başar, "Batch-to-batch finite-horizon LQ control for unknown discrete-time linear systems via stochastic extremum seeking," *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 4116–4123, 2017.

[193] B. Pang, T. Bian, and Z.-P. Jiang, "Data-driven finite-horizon optimal control for linear time-varying discrete-time systems," in *IEEE Conference on Decision and Control*, 2018, pp. 861–866.

[194] S. Baros, C.-Y. Chang, G. E. Colón-Reyes, and A. Bernstein, "Online data-enabled predictive control," *Automatica*, vol. 138, p. 109926, 2022.

[195] M. Verhaegen and X. Yu, "A class of subspace model identification algorithms to identify periodically and arbitrarily time-varying systems," *Automatica*, vol. 31, no. 2, pp. 201–216, 1995.

[196] A. Ichikawa and H. Katayama, "Linear time varying systems and sampled-data systems," in *Lecture Notes in Control and Information Sciences*, M. Thoma and M. Morari, Eds. Springer-Verlag, 2001, vol. 265.

[197] D. P. Bertsekas, *Dynamic programming and optimal control*, 3rd ed. Athena Scientific, 2005, vol. 1.

[198] A. Gattami, "Generalized linear quadratic control," *IEEE Transactions on Automatic Control*, vol. 55, no. 1, pp. 131–136, 2010.

[199] M. Rotulo, C. De Persis, and P. Tesi, "Data-driven linear quadratic regulation via semidefinite programming," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 3995–4000, 2020.

[200] Z.-P. Jiang and Y. Wang, "Input-to-state stability for discrete-time nonlinear systems," *Automatica*, vol. 37, no. 6, pp. 857–869, 2001.

[201] C. Scherer and S. Weiland, *Linear Matrix Inequalities in Control*, Jan. 2015.

[202] C. W. Scherer, "Robust mixed control and linear parameter-varying control with full block scalings," in *Advances in Linear Matrix Inequality Methods in Control*. SIAM, Jan. 2000, pp. 187–207.

[203] J. M. Fry, M. Farhood, and P. Seiler, "IQC-based robustness analysis of discrete-time linear time-varying systems," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 16, pp. 3135–3157, 2017.

[204] C. Byrnes and W. Lin, "Losslessness, feedback equivalence, and the global stabilization of discrete-time nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 39, no. 1, pp. 83–98, 1994.

[205] M. Green and D. J. Limebeer, *Linear Robust Control*. Dover Publications, 2012.

[206] M. Sznaier, T. Amishima, P. Parrilo, and J. Tierno, "A convex approach to robust h_2 performance analysis," *Automatica*, vol. 38, no. 6, pp. 957–966, 2002.

[207] D. V. Balandin, R. S. Biryukov, and M. M. Kogan, "Finite-horizon optimal control of maximal output deviations in ltv discrete-time systems," in *IEEE Mediterranean Conference on Control and Automation*, 2019, pp. 142–147.

[208] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, 2014.

[209] J.-M. Rodriguez-Bernuz, I. McInerney, A. Junyent-Ferre, and E. C. Kerrigan, "Design of a linear time-varying model predictive control energy regulator for grid-tied VSCs," *IEEE Transactions on Energy Conversion*, vol. 36, no. 2, pp. 1425–1434, 2021.

[210] J. Gray, "The mechanism of locomotion in snakes," *Journal of Experimental Biology*, vol. 23, no. 2, pp. 101–120, 1946.

[211] S. Hirose, *Biologically inspired robots.* Oxford Univ. Press, 1993.

[212] P. Liljebäck, K. Y. Pettersen, Ø. Stavdahl, and J. T. Gravdahl, "A simplified model of planar snake robot locomotion," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 2868–2875.

[213] V. Balakrishnan and L. Vandenberghe, "Semidefinite programming duality and linear time-invariant systems," *IEEE Transactions on Automatic Control*, vol. 48, no. 1, pp. 30–41, 2003.

[214] E. Feron, V. Balakrishnan, S. Boyd, and L. El Ghaoui, "Numerical methods for H_2 related problems," in *American Control Conference*, vol. 4, 1992, pp. 2921–2922.

[215] D. A. Braun, P. A. Ortega, and D. M. Wolpert, "Nash equilibria in multi-agent motor interactions," *PLoS Computational Biology*, vol. 5, no. 8, p. e1000468, 2009.

[216] G. F. Franklin, J. D. Powell, and M. L. Workman, *Digital Control of Dynamic Systems*, 3rd ed. Addison-Wesley, 1998.

[217] M. Sassano, T. Mylvaganam, and A. Astolfi, "An algebraic approach to dynamic optimisation of nonlinear systems: a survey and some new results," *Journal of Control and Decision*, vol. 6, no. 1, p. 1–29, 2018.

[218] T. Mylvaganam and A. Astolfi, "Approximate optimal monitoring," in *European Control Conference*, 2014, pp. 1199–1204.

[219] J. Peng, Y. Zhu, W. Shu, and M.-Y. Wu, "When data contributors meet multiple crowdsourcers: Bilateral competition in mobile crowdsourcing," *Computer Networks*, vol. 95, p. 1–14, 2016.

[220] V. S. Dasari, B. Kantarci, M. Pouryazdan, L. Foschini, and M. Girolami, "Game theory in mobile CrowdSensing: A comprehensive survey," *Sensors*, vol. 20, no. 7, p. 2055, 2020.

[221] Z. Zheng, L. Song, Z. Han, G. Y. Li, and H. V. Poor, "Game theory for Big Data processing: Multileader multifollower game-based ADMM," *IEEE Transactions on Signal Processing*, vol. 66, no. 15, pp. 3933–3945, 2018.

[222] B. Faltings and G. Radanovic, *Game Theory for Data Science: Eliciting Truthful Information.* Springer International Publishing, 2017.

[223] A. Bonatti, M. Dahleh, T. Horel, and A. Nouripour, "Coordination via selling information," *arXiv preprint:* 10.48550/arXiv.2302.12223, 2023.

[224] B. Nortmann, A. Monti, M. Sassano, and T. Mylvaganam, "Nash equilibria for linear quadratic discrete-time dynamic games via iterative and data-driven algorithms," *IEEE Transactions on Automatic Control*, 2024, ([BN1]).

[225] B. Nortmann and T. Mylvaganam, "Direct data-driven control of linear time-varying systems," *IEEE Transactions on Automatic Control*, vol. 68, no. 8, pp. 4888–4895, 2023, ([BN2]).

[226] B. Nortmann, A. Monti, M. Sassano, and T. Mylvaganam, "Feedback Nash equilibria for scalar two-player linear-quadratic discrete-time dynamic games," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 1772–1777, 2023, ([BN3]).

[227] B. Nortmann and T. Mylvaganam, "Approximate Nash equilibria for discrete-time linear quadratic dynamic games," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 1760–1765, 2023, ([BN4]).

[228] B. Nortmann, A. Monti, T. Mylvaganam, and M. Sassano, "Nash equilibria for scalar LQ games: iterative and data-driven algorithms," in *IEEE Conference on Decision and Control*, 2022, pp. 3801–3806, ([BN5]).

[229] B. Nortmann and T. Mylvaganam, "Data-driven cost representation for optimal control and its relevance to a class of asymmetric linear quadratic dynamic games," in *European Control Conference*, 2022, pp. 2185–2190, ([BN6]).

[230] ——, "Data-driven control of linear time-varying systems," in *IEEE Conference on Decision and Control*, 2020, pp. 3939–3944, ([BN7]).