NORTHWESTERN UNIVERSITY

Analysis and Design of Algorithms for Dynamic Average Consensus and Convex Optimization

A DISSERTATION

SUBMITTED TO THE GRADUATE SCHOOL
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS

for the degree

DOCTOR OF PHILOSOPHY

Field of Electrical Engineering

By

Bryan R. Van Scoy

EVANSTON, ILLINOIS

June 2017

© Copyright by Bryan R. Van Scoy 2017

All Rights Reserved

Abstract

Analysis and Design of Algorithms for Dynamic Average Consensus and Convex Optimization

Bryan R. Van Scoy

Algorithms which are efficient and robust are essential to meet the increasing computational demands in the world today. In this thesis, we consider the analysis and design of both distributed algorithms for dynamic average consensus and centralized algorithms for convex optimization.

Dynamic average consensus consists of a group of agents, each with a local signal (such as the output of a sensor), where the goal is for each agent to use communication with local neighbors to estimate the global average of the signals. We analyze and design diffusion algorithms based on local averaging to solve the dynamic average consensus problem for two classes of signals. First, we consider signals whose frequency spectrum is nonzero only at a finite number of discrete frequencies. In this case, the local agent dynamics use feedback to converge to the exact average of the signals. Furthermore, we design estimators which (1) are scalable to a large number of agents, (2) are robust to both initialization errors and agents entering/leaving the group, (3) have internally stable dynamics, (4) use discrete-time local broadcast communication, and (5) have fast convergence rates. The second class of signals

considered are bandlimited; that is, their frequency spectrum is nonzero in a continuous band of frequencies. The feedback estimators designed previously are shown to be inadequate in this case. Instead, we propose a feedforward estimator which can track the average with arbitrarily small error under very limited assumptions, including when the communication among the agents changes rapidly in time.

Dynamic average consensus can be formulated as a convex optimization problem, and we show that the analysis and design of algorithms for the two problems are quite similar. In the centralized setting, we design and analyze a novel gradient-based algorithm for unconstrained convex optimization. When the objective function is strongly convex with strong convexity parameter m, and its gradient is Lipschitz continuous with Lipschitz constant L, then the iterates and function values converge linearly to the optimum at rates ρ and ρ^2, respectively, where $\rho = 1 - \sqrt{m/L}$. These are the fastest known guaranteed linear convergence rates for globally convergent first-order methods, and for high desired accuracies the corresponding iteration complexity is within a factor of two of the theoretical lower bound. We use a simple graphical design procedure based on integral quadratic constraints to derive closed-form expressions for the algorithm parameters. The new algorithm, which we call the triple momentum method, can be seen as an extension of gradient descent, Nesterov's accelerated gradient descent, and the heavy-ball method.

Acknowledgments

This thesis is the culmination of not only my efforts while at Northwestern, but also all those who have poured into me and helped me along the way. With all the guidance and encouragement that I received, this road has been quite enjoyable for which I am truly thankful.

First of all, I would like to thank my advisor, Randy Freeman. He has guided me through every step of this process, from choosing to come to Northwestern, to writing my first research paper, and now in completing my PhD. His door has always been open for me to ask questions, and his diligence to produce exceptional work has kept me striving to improve my own capabilities.

I would also like to thank the others in my research group, Kevin Lynch, Matthew Elwin, Jemin George, Daniel Burbano, and the others who have passed through. Hearing what they had been working on each week has broadened my understanding, and their feedback along the way was immensely valuable.

My other committee members, Randy Berry, Ermin Wei, and Jorge Cortés, have also been instrumental in providing feedback and helping to produce this work.

While those within the academic world have played a crucial role in finishing this thesis, an equally important role was played by those who have been in my life encouraging me along the way. My mom and dad have helped me pursue my dreams from the very beginning. Too many friends to name have been with me through this whole experience, from my roommates

Greg Kimmel, George Kacouris, and Mark Harmon to my brothers and sisters at Evanston Bible Fellowship including Vlad Serban, Jacob DiEdwardo, Cristina Hunter, and so many others. They pushed me to work hard, but also pulled me away from research to remind me that there is more to life than just math.

Table of Contents

List of Tables

List of Figures

CHAPTER 1

Introduction

Algorithms which are efficient and robust are essential to meet the increasing computational demands in the world today. Applications in distributed multi-agent systems as well as centralized systems require algorithms with provable guarantees on properties such as convergence rate, steady-state error, robustness to initial conditions, etc. In order to develop such algorithms we must not only be able to analyze existing algorithms, but also have methods of designing new algorithms to achieve the required specifications. In this thesis, we consider the analysis and design of both distributed algorithms for dynamic average consensus and centralized algorithms for convex optimization.

1.1. Dynamic Average Consensus

Consider a group of agents where each agent has a time-varying local input signal and is capable of local communication with neighboring agents in a communication network which may change over time. The dynamic average consensus problem is for each agent to maintain a running local estimate of the current global average of the time-varying input signals. Hence we say that the dynamic average consensus algorithm running on each agent is an *estimator*.

Estimators for dynamic average consensus have many applications in the decentralized control of multi-agent systems, and are fundamental building blocks of many other distributed algorithms. Some distributed applications include

- environmental monitoring [**1, 2, 3, 4, 5, 6**],
- sensor fusion [**7**],
- formation control [**8**],
- dynamic merging of feature-based maps [**9**], and
- optimization [**10**].

We consider diffusive approaches to this problem which are inherently scalable, distributed, indifferent to network structure, and frugal in their use of agent memory, computation, and communication resources. Diffusive estimators spread information through the network of agents using local averaging. With diffusive algorithms, agents can average internal variables with their local neighbors as well as perform any necessary internal calculations.

Estimators for dynamic average consensus are designed to have desired properties such as robustness and fast convergence. In particular, we design estimators to have the following properties:

Exactness: Under constant inputs, all agent outputs converge to the exact global average of the input signals.

Initialization robustness: The initial values of the internal agent states have no effect on the steady-state estimates.

Time-invariance: The estimator dynamics are time-invariant (in particular, transient characteristics like settling time do not change as the estimator runs).

Internal stability: The estimator dynamics are bounded-input, bounded-state stable.

Fast convergence: The estimates converge to steady state quickly.

Robustness to changes in the graph: The estimates have the same properties when the communication graph is time-varying as when it is constant.

Discrete time updates: Local communication with neighboring agents occurs at discrete time steps.

Local broadcast communication: At each discrete-time update, each agent broadcasts a single message to all of its immediate neighbors.

Exactness is desirable not only because it guarantees zero steady-state error for constant inputs, but also because it leads to small steady-state error for slowly-varying inputs. Initialization robustness makes it easy for the network to recover from changes in the network topology, temporary asymmetries in communication, or the addition or deletion of agents; the estimators can just keep running as usual and no special action is required. Robustness to changes in the communication graph is an important property of a distributed estimator, as network changes can occur due to

- mobile agents with range-limited communication,
- noisy communication in which packets are dropped at random,
- agent failure, and
- addition of agents to the network.

We now show how our work relates to the vast literature on dynamic average consensus.

1.1.1. Literature Review

If the agent inputs are constants, then their average can be computed using a variety of diffusive methods in which the inputs are used as initial states and the agent outputs converge to the average of the initial states. Such methods are known as *static average consensus* and have been studied extensively since the work of Tsitsiklis in 1984 [**11**].

Since static average consensus algorithms were first introduced, a big concern has been optimizing the convergence rate. When the topology of the communication graph is known and the edge weights can be chosen in a centralized manner, semidefinite programming can be used to optimize the convergence rate if the graph is undirected [**12**]. Besides optimizing the edge weights, the convergence rate can also be improved through the use of a local predictor at each agent [**13**]. Another approach to improve convergence is to use the alternating direction method of multipliers (ADMM) and distributed optimization to select optimal parameters [**14**].

Static estimators calculate the average of the inputs, but are incapable of *tracking* the average of a time-varying input. Furthermore, they are inherently not robust to communication faults since a single dropped packet can cause the estimator to converge to an incorrect value [**15**]. In 2005, Spanos et al. [**16**] introduced *dynamic average consensus* in which the agent inputs appear explicitly in the update equations themselves, not just as initial states. In the dynamic case, agent inputs can change with time without the need for any reinitialization, and each agent maintains a running local estimate of the current global average of the input signals.

Dynamic average consensus in continuous time has also received much attention [**17, 18**]. Estimators in continuous time can converge to the exact average of the time-varying input signals in finite time so long as the derivatives of the input signals are bounded [**18**]. However, this requires communication between neighboring agents to occur continuously which is not realistic in many scenarios. Furthermore, many continuous-time estimators use high-gain approaches which require very small step sizes when discretized. Therefore, we focus on the more practical scenario when communication occurs at discrete time steps.

Many diffusive estimators for dynamic average consensus in discrete time have been proposed [**16, 19, 20, 21, 22, 23, 24, 25**]. Each estimator has different properties based on the assumptions made on both the communication graph and the input signals.

As a first step to tracking time-varying inputs, the static estimator was modified so that the input signals enter as the inputs of a dynamical system, not as the initial state [**16**]. Since the input is no longer the initial state, estimators are referred to as *robust to initial conditions* if the steady-state value is independent of the initial state. Estimators which are robust to initial conditions can track slowly-varying inputs and can recover from changes in the communication graph. Furthermore, zero steady-state error can be achieved if the feedback loop contains an integrator.

The estimators in [**23, 26, 27**] lack initialization robustness and consequently must take care to correctly handle these situations without introducing additional errors in steady state. For example, [**23**] suggest that an agent send a particular message to its neighbors upon leaving the network, but an abruptly failing agent is unlikely to send such a message. Therefore, we design estimators which are robust to initial conditions.

Several techniques have been proposed to make dynamic estimators which are robust to initial conditions. A simple method is to perform the diffusion step *after* the integrator in the feedback path. However, if only a single variable is transmitted to neighboring agents at each iteration, then this method sacrifices stability of the internal agent states [**20**].

To have both robustness to initial conditions and internal stability, we could use multiple rounds of communication per iteration to produce the *polynomial filter estimator*. Static versions of this estimator were first designed numerically in [**28**], and closed-form expressions in terms of Chebyshev polynomials were given in [**29**]. We extend this to time-varying signals by producing an equivalent dynamic estimator, although the estimator is not robust to initial conditions. With a simple modification, however, we can make the estimator robust. The resulting polynomial filter estimator is both robust to initial conditions and internally stable. We also show that, for estimators which are not robust to initial conditions, the optimal convergence rate is achieved by an estimator with two internal state variables which uses one round of communication per iteration. The cost of making the estimator robust to initial conditions is that the convergence rate is slower for any finite number of rounds of communication per iteration, although the same convergence rate as the non-robust case is obtained in the limit as the number of rounds of communication per iteration approaches infinity.

Another method of making an estimator robust to initial conditions is to use time-varying dynamics [**27**]. This exploits a tradeoff between convergence rate and robustness to initial conditions. The approach is attractive in theory since it is robust to initial conditions and achieves zero steady-state error, but does not work well in practice. In the beginning phase the estimator recovers from changes in the graph quickly, but the error is large. As the estimator dynamics change, the estimator takes longer and longer to recover from changes in the graph; robustness slowly deteriorates while the steady-state error is improved, but at any finite time there is a trade-off between the two properties.

A time-invariant approach which is both robust to initial conditions and internally stable is to apply both proportional and integral terms in the feedback loop [**19**]. The resulting estimator, called the *proportional-integral (PI) estimator*, has convergence rate which is quite slow compared to other methods. Elwin et al. [**30, 31**] optimize the worst-case convergence rate by applying global numerical optimization solvers, but no guarantees are given of finding the global optimum. To fix this issue, we first calculate closed-form solutions for the estimator parameters which optimize the convergence rate. We then show that extra dynamics can be introduced on each agent to further increase the convergence rate, and give closed-form expressions for the optimal parameters. This requires each agent to have four internal state variables instead of just two, but still uses the same amount of communication (each agent must broadcast two state variables in the same packet to neighboring agents at each iteration).

The PI estimator has another desireable property in that it can track the average of the input signals even when the communication graph changes randomly at each iteration. When the graph is time-varying, the estimator outputs converge to a random signal whose time average is equal to its statistical average. We call this property *asymptotic mean ergodicity*. In this case, each agent can apply a local lowpass filter to recover the average of the input signals. Others have studied the properties of static estimators over time-varying graphs [**32, 33, 34, 35, 36**], but our result applies to general dynamic estimators.

Another method of achieving both robustness to initial conditions and internal stability is to factor the graph Laplacian matrix and move the integrator between each of the factors. We call the resulting algorithm the *edge estimator* because the number of states in the system corresponds to the number of edges in the communication graph instead of the number of agents. To implement this estimator, each agent must maintain an internal state variable for each agent from which it receives information. This approach is not scalable to large dense networks, but can be used if each agent has sufficient memory to store the required number of variables.

The last approach we consider to obtaining robustness to initial conditions and internal stability is to change the state-space of the estimator. We start with the form of the proportional estimator which is robust to initial conditions but not internally stable. We then change the state-space of the integrator states from Euclidean space to the multi-dimensional torus (which is compact). The result is that the integrator states are inherently bounded, so both properties are achieved. The nonlinear state-space requires nonlinear dynamics in the estimator. To do this, we generalize the Laplacian matrix of a graph to the nonlinear Laplacian operator. This type of estimator was studied in the static case in [**37**] and [**38**, Chapter 2]. Our dynamic estimator has zero steady-state error if the initial state is located within an open region of attraction. In other words, the estimator is *locally* convergent but not *globally* convergent. For linear estimators these properties are equivalent, and we refer to an estimator which is globally convergent as robust to initial conditions. In the nonlinear case, however, we must distinguish between local and global convergence. Although the nonlinear estimator only has local convergence, its convergence rate is equivalent to the fastest known methods.

Dynamic estimators which are robust to initial conditions and contain an integrator in the feedback loop can track slowly-varying inputs and can recover from changes in the communication graph after a transient. They are restricted to slowly-varying inputs, however, since they are designed to track constant inputs with zero steady-state error.

Several steps have been taken towards tracking fast time-varying signals. To accommodate arbitrarily fast time-varying input signals with a known model (or frequency), the internal model estimator was introduced [**16, 21**]. Instead of an integrator, the internal model of the input is placed in the feedback loop causing the estimator to have zero steady-state error for input signals at that frequency. This estimator can track arbitrarily fast time-varying signals, but the model of the input must be known a priori to design the estimator. In the case when the signal frequency is unknown, it was shown that the frequency can be estimated and the estimate used in the internal model estimator [**22**]. This method also achieves zero steady-state error, but the signal must be composed of a single frequency. Another approach is to design the estimator using a very general internal model so that it applies to a large set of input signals. This approach is taken in [**23**] where an estimator is designed using a fixed degree polynomial as the internal model. The resulting estimator can track input signals whose derivatives are bounded with finite steady-state error.

All of the previous approaches rely on using a model (or approximate model) of the input signals in the feedback loop. This implies that the input signals have such a model, which is only the case if their frequency spectrum is nonzero at a finite number of *discrete* frequencies. In practice, the input signals will often come from the output of sensors after applying a lowpass filter. Therefore, we also consider the case when the input signals have a *continuous* frequency spectrum (e.g., the signals are bandlimited).

To track the average of input signals with continuous frequency spectrum, we propose a *feedforward estimator* as opposed to the feedback estimators mentioned previously. When the input signals are bandlimited with known cutoff frequency, the estimates track the global average of the input signals with bounded error and no delay, i.e., the estimate of the global average is known on each agent at every iteration and it approximates the average of the inputs at that time instant. Furthermore, the proposed estimator is robust to changes in the graph, meaning that the performance is no worse when the graph is time-varying than when it is constant. Distinct characteristics of the design are that the estimator applies the Laplacian in the feedforward path, and each agent applies a prefilter to the input signal *before* any communication with neighbors. The estimator can be decomposed into two components:

a bandlimited prediction filter followed by a feedforward averaging filter. A bandlimited prediction filter with finite impulse response (FIR) is given in [**39**]. As an alternative, we design an infinite impulse response (IIR) filter. We show how to design both filters to minimize the error in the prediction while taking into account the precision of the arithmetic used to implement the filter. In particular, we do not assume exact precision arithmetic. Using either of these filters, we show that the steady-state error of the feedforward estimator can be made arbitrarily small if (1) enough state variables are used on each agent, (2) enough variables are transmitted in each packet, and (3) exact arithmetic is used. The error is finite when using finite-precision arithmetic and a finite number of state variables and transmitted variables. However, we show that the error using our IIR filter is smaller than when using the FIR filter, although the transient response takes longer to decay.

There is a vast amount of literature related to the dynamic average consensus problem, and we have presented only a brief review. A few examples of topics not covered in this thesis are

- privacy of estimator inputs and internal states [**24, 40**],
- estimators for unbalanced directed graphs [**41, 42**],
- consensus of quantized values, [**43, 44**] and
- event-triggered estimators [**45, 46**].

1.1.2. Contributions

To summarize, our main contributions to the dynamic average consensus problem are as follows:

- We characterize the asymptotic mean ergodicity property of estimators. For one-hop estimators with this property, each agent can apply a local lowpass filter to recover the average of the inputs when the communication graph is time-varying.
- Closed-form solutions are found for the PI estimator parameters which optimize the convergence rate. We also show that additional dynamics can be introduced into the PI estimator to achieve even faster convergence and give the optimal parameters in that case as well.
- We design and analyze a nonlinear estimator which has the following properties: 1) exact, 2) scalable, 3) internally stable, 4) time-invariant, 5) one-hop local broadcast communication, and 6) locally convergent with convergence rate equivalent to the fastest known methods.
- We design and analyze a feedforward estimator to solve the dynamic average consensus problem when the input signals are bandlimited and the communication graph is time-varying. The estimator can achieve arbitrarily small steady-state error, even when the graph changes at each iteration.
- We design and analyze an IIR bandlimited prediction filter. We also show how to minimize the error in the prediction while taking into account the precision arithmetic used for both the FIR and IIR filters. If exact arithmetic is used, both filters can predict the signal with arbitrary accuracy so long as the cutoff frequency is strictly less than the Nyquist frequency. Using finite-precision arithmetic, our IIR filter has smaller steady-state error but a longer transient than the FIR filter.

The design and analysis of dynamic average consensus estimators is very similar to that of gradient-based algorithms for the unconstrainted optimization of convex functions. This problem is described in the following section.

1.2. Convex Optimization

Consider the optimization problem

$$\underset{x \in \mathbb{R}^n}{\text{minimize}} \;\; f(x) \tag{1.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, strongly convex with parameter m, and has a Lipschitz continuous gradient with Lipschitz constant L. Since f is strongly convex, it has a unique global minimizer $x_\star \in \mathbb{R}^n$. We consider first-order (gradient-based) algorithms to solve (1.1).

Perhaps the simplest algorithm which solves (1.1) is gradient descent with constant step size, which has the form

$$x_{k+1} = x_k - \alpha \nabla f(x_k), \quad x_0 \in \mathbb{R}^n.$$

Using $\alpha = 2/(L+m)$, the iterates converge globally and linearly to the optimizer with rate $(L-m)/(L+m)$.[1]

Due to the slow convergence of gradient descent, many methods have been proposed to obtain faster convergence. In general, faster convergence rates can be achieved by introducing *momentum*. Examples of methods which incorporate momentum include the heavy-ball method [**47**],

$$x_{k+1} = (1+\beta)x_k - \beta x_{k-1} - \alpha \nabla f(x_k),$$

[1] Throughout the thesis, the phrase "linear convergence with rate ρ" means R-linear convergence, i.e., having errors bounded by $c\rho^k$ for some constant $c > 0$ and for all $k \geq 0$.

and Nesterov's accelerated gradient descent [**48**],

$$x_{k+1} = y_k - \alpha \nabla f(y_k)$$
$$y_k = (1+\beta) x_k - \beta x_{k-1}.$$

Although heavily studied, it remains an open question how to choose the parameters α and β in order to achieve global convergence while optimizing the convergence rate. For the heavy-ball method, one can choose parameters to achieve a *local* convergence rate of $(\sqrt{L} - \sqrt{m})/(\sqrt{L} + \sqrt{m})$, but the resulting method does not converge globally [**49**]. For other parameter choices, the heavy-ball method converges globally to the optimizer with a linear rate, although a tight bound on the rate has not been found [**50**].

Other attempts have been made at developing algorithms with optimal convergence rates. Regularized update descent [**51**] generalizes the heavy-ball and Nesterov's method, but the convergence rate has not been found. The convergence rate of a general multi-step method was optimized numerically, but no closed-form solution was found [**49**]. The optimal parameters can be found by solving semidefinite programs [**52**], and closed-form solutions for the time-varying parameters were found for the case when f is weakly-convex [**53**].

In his book [**48**], Nesterov gives several choices of both constant and time-varying parameters which guarantee that the function values generated by his algorithm converge with linear rate $1-\sqrt{m/L}$ if f is strongly convex and sublinearly as $\mathcal{O}(1/k^2)$ if f is weakly convex.[2] The derived bound on the corresponding iteration complexity (i.e., the number of iterations required to minimize the objective function to within a given tolerance) is proportional to a theoretical lower bound, so his method is often called optimal [**48**, Thm. 2.2.2]. It has

[2] Throughout the thesis, "weakly convex" means convex but not necessarily strongly convex.

recently been shown, however, that for weakly convex objective functions, other algorithms can achieve bounds on the iteration complexity that are about a factor of $\sqrt{2}$ smaller than the bound for Nesterov's method [**52, 53**]. Similarly, in this paper we achieve a reduction of a factor of about two for the class of strongly convex objective functions.

1.2.1. Contributions

We develop a novel algorithm to solve (1.1) when the objective function is m-strongly convex and its gradient is L-Lipschitz continuous with m and L known. Our algorithm, called the triple momentum (TM) method, uses three momentum terms to achieve global linear convergence to the optimizer. The iterates and function values converge linearly to the optimum at rates ρ and ρ^2, respectively, where $\rho = 1 - \sqrt{m/L}$. These are the fastest known guaranteed linear convergence rates for globally convergent first-order methods. For high desired accuracies the corresponding iteration complexity is a factor of two better than Nesterov's method and within a factor of two of the theoretical lower bound. The new algorithm can be seen as an extension of methods such as gradient descent, Nesterov's accelerated gradient descent, and the heavy-ball method.

Inspired by [**49**], we use integral quadratic constraints from robust control to both *analyze* and *design* our algorithm. A simple graphical design procedure based on integral quadratic constraints is used to derive closed-form expressions for the constant algorithm parameters. We also provide an alternative convergence proof of the triple momentum method which does not use control theory.

1.3. Notation

Linear algebra: Define the $n \times 1$ vectors $\mathbf{1}_n$ and $\mathbf{0}_n$ of all ones and zeros, respectively, and I_n as the $n \times n$ identity matrix. Denote $\Pi_n = \mathbf{1}_n \mathbf{1}_n^T / n$. The spectral radius is denoted $\rho(\cdot)$. The transpose of A is denoted by A^T. The Moore-Penrose pseudoinverse of a matrix A is denoted $A^\dagger$. A diagonal matrix with entries α_i on the diagonal is denoted $\operatorname{diag}(\alpha)$. $A \succ 0$ and $A \succeq 0$ mean that the matrix A is positive definite and positive semidefinite, respectively. The Kronecker product is denoted $\otimes$. Denote the set of real symmetric $n \times n$ matrices as $\mathbb{S}^n$. For a matrix, $\operatorname{cond}(\cdot)$ is the condition number and $\sigma_{\min}(\cdot)$ is the minimum singular value.

Norms and sequences: The 2-norm is denoted $\| \cdot \| : \mathbb{R}^n \to \mathbb{R}$. Define ℓ_{2e}^n as the set of all one-sided sequences $x : \mathbb{N} \to \mathbb{R}^n$. The subset $\ell_2^n \subset \ell_{2e}^n$ consists of all square-summable sequences, i.e., $x \in \ell_2^n$ if and only if $\sum_{k=0}^{\infty} \|x_k\|^2 < \infty$. The limit superior of a sequence is $\limsup_{k\to\infty} c_k = \lim_{k\to\infty} \sup_{j \geq k} c_j$. The term linear convergence is used to denote R-linear convergence, i.e., we say the sequence x_k converges linearly to $x_\star$ with rate ρ if $\|x_k - x_\star\| \leq c\rho^k$ for all $k \geq 0$ with $c > 0$.

Statistics: The expectation of a random variable x is denoted $\mathrm{E}[x]$, and the expectation of a matrix A is denoted $\mathrm{E}[A]$ and is the element-wise expectation of each element.

Complex numbers: Denote the n-dimensional complex plane by $\mathbb{C}^n$. The unit circle in the complex plane is denoted $S^1 = \{z \in \mathbb{C} : |z| = 1\}$, and the n-dimensional torus is $\mathbb{T}^n = S^1 \times \cdots S^1$ (n times).

Linear systems: The z-transform of a sequence $x \in \ell_{2e}$ is denoted $x(z) = \sum_{k=-\infty}^{\infty} x_k \, z^{-k}$. A linear system $G : \ell_{2e} \to \ell_{2e}$ defines the map $y = G(u)$ which describes the recursion

$$\begin{aligned} \xi_{k+1} &= A\xi_k + Bu_k \\ y_k &= C\xi_k + Du_k \end{aligned}$$

where ξ_0 is the initial condition. The transfer function (with zero initial conditions) is $G(z) = C(zI - A)^{-1}B + D$, and we write

$$G = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right].$$

For block systems, we use the notation

$$G \otimes I = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \otimes I = \left[\begin{array}{c|c} A \otimes I & B \otimes I \\ \hline C \otimes I & D \otimes I \end{array}\right].$$

CHAPTER 2

Dynamic Average Consensus

In this chapter we setup the dynamic average consensus problem. First, we describe the problem along with the various assumptions we will make on both the communication among the agents and the signals which they are tracking. We then discuss the different properties that we would like estimators to have. To represent estimators for dynamic average consensus, we give both block diagram and state-space forms. We then present some basic analyais tools such as the separated system and the palindromic transformation that will be used to design and analyze the estimators.

2.1. Problem Definition

Consider a group of n agents where each agent has a discrete-time local scalar input signal. The input at time k at agent i is denoted u_k^i. The discrete-time signal can be obtained from a continuous-time signal by sampling and then applying an anti-aliasing filter to obtain a discrete-time bandlimited signal as shown in Fig. 2.1. Each agent runs an estimator with state $x_k^i \in \mathbb{R}^p$ using its own local input signal along with information from its neighbors to produce a local scalar output signal y_k^i. The dynamic average consensus problem is for the output of each agent to track the global average of the input signals. We define the error at agent i at time k to be the difference between the output of the agent and the global average of the input signals at the same time instant,

$$e_k^i = y_k^i - \frac{1}{n}\sum_{i=1}^{n} u_k^i. \tag{2.1}$$

We also define an iteration to be the computations performed when each agent can broadcast no more than a single packet of information to its local neighbors, i.e., single-hop communication is used. To use compact notation, the variables associated to each agent are stacked to form vectors,

$$u_k = \begin{bmatrix} u_k^1 \\ \vdots \\ u_k^n \end{bmatrix}, \qquad x_k = \begin{bmatrix} x_k^1 \\ \vdots \\ x_k^n \end{bmatrix}, \qquad y_k = \begin{bmatrix} y_k^1 \\ \vdots \\ y_k^n \end{bmatrix}, \qquad e_k = \begin{bmatrix} e_k^1 \\ \vdots \\ e_k^n \end{bmatrix}. \tag{2.2}$$

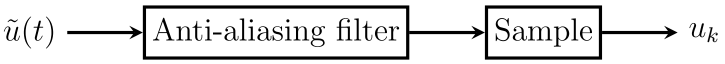


Figure 2.1. Obtaining a discrete-time signal from a continuous-time signal. The continous-time signal $\tilde{u}(t)$ is passed through an anti-aliasing filter and sampled to obtain a discrete-time signal u_k.

The dynamic average consensus problem can be described as follows.

Problem 1 (Distributed algorithm design)**.** *Given:*

(1) assumptions on the input signals

(2) assumptions on the communication among agents

(3) desired properties of the algorithm

Design a distributed algorithm such that, under the given assumptions on the input signals and the communication among agents, the algorithm satisfies the desired properties.

Desired properties of estimators for dynamic average consensus include requirements on the error in (2.1). Some examples include

- the steady-state error is either zero or bounded,
- the error converges at a specified rate (e.g., linear, quadratic, finite time),
- the steady-state error is independent of the initial conditions.

Other algorithm properties that we consider are the following:

- the memory and computations required to implement the algorithm scale with the number of agents,
- the internal estimator states remain bounded for all time if the input signals are bounded.

The communication among the agents is represented by a graph, as described in the following section.

2.2. Graph Theory

The communication topology is modeled as a weighted directed graph given by the triple $\mathcal{G} = (\mathcal{V}, \mathcal{E}, A)$, where $\mathcal{V} = \{1, \ldots, n\}$ is the node set, $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ is the edge set, and $A = [a_{ij}] \in \mathbb{R}^{n \times n}$ is the weighted adjacency matrix. We associate each node in the graph with an agent, and the directed links in the graph as communication channels between agents. We label and number the edges, writing $\mathcal{E} = \{\mathfrak{e}_1, \ldots, \mathfrak{e}_m\}$ where $m = |\mathcal{E}|$. There is a directed edge from node i to node j, i.e., $(i, j) \in \mathcal{E}$, if agent i receives information from agent j, and the edge weight is a_{ij}. By convention, $(i, i) \notin \mathcal{E}$. We set $a_{ij} = 0$ if agent i does not receive information from agent j, i.e., $(i, j) \notin \mathcal{E}$. The *in-neighbors* of node i, denoted $\mathcal{N}_{\text{in}}(i)$, is the set of nodes such that $(i, j) \in \mathcal{E}$ for some $j \in \mathcal{V}$. The *out-neighbors* of node i, denoted $\mathcal{N}_{\text{out}}(i)$, is the set of nodes such that $(j, i) \in \mathcal{E}$ for some $j \in \mathcal{V}$. The *weighted in-degree* and *weighted* *out-degree* of node i are $d_{\text{in}}(i) = \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}$ and $d_{\text{out}}(i) = \sum_{j \in \mathcal{N}_{\text{out}}(i)} a_{ji}$, respectively. A *directed path* is a sequence of nodes $i_1, i_2, \ldots, i_p$ such that $(i_\ell, i_{\ell+1}) \in \mathcal{E}$ for $\ell = 1, \ldots, p-1$. The graph is *strongly connected* when there is a directed path between any pair of nodes. The graph is called *undirected* if $a_{ij} = a_{ji}$ for all $i, j \in \mathcal{V}$.

The Laplacian matrix is $L = \text{diag}(A\mathbf{1}_n) - A$ which satisfies $L\mathbf{1}_n = \mathbf{0}_n$. The graph is said to be *balanced* if the weighted in-degree of each node is equal to its weighted out-degree, which is equivalent to the condition $\mathbf{1}_n^T L = \mathbf{0}_n^T$. The Laplacian matrix can be decomposed by its eigenvalues and eigenvectors as $L = V\Lambda V^{-1}$ where $V = [v_1, \ldots, v_n]$ and $\Lambda = \text{diag}(\lambda_1, \ldots, \lambda_n)$. For $i = 1, \ldots, n$, v_i is an eigenvector of L with eigenvalue λ_i. Without loss of generality, we assume that $v_1 = \mathbf{1}_n/\sqrt{n}$ and $\lambda_1 = 0$. It can be shown that the eigenvalues of the Laplacian have nonnegative real parts, and we order them by $\text{Re}\{\lambda_i\} \leq \text{Re}\{\lambda_j\}$ for $i < j$. Zero is a simple eigenvalue of L for strongly connected graphs.

For undirected graphs, the Laplacian matrix is symmetric ($L = L^T$) so the eigenvectors are orthogonal and the eigenvalues are real, i.e., $V^TV = VV^T = I_n$ and $\text{Im}\{\lambda_i\} = 0$ for $i \in \{1, \ldots, n\}$. Furthermore, the Laplacian matrix of an undirected graph can be decomposed as $L = BWB^T$ where $B \in \{-1, 0, 1\}^{n \times m}$ is the oriented incidence matrix for the graph and $W \in \mathbb{R}^{m \times m}$ is a diagonal weight matrix. Specifically, B is defined as

$$B_{i\ell} = \begin{cases} 1 & \text{if } i \in \mathfrak{e}_\ell \text{ and } i = \min\{j \,:\, j \in \mathfrak{e}_\ell\} \\ -1 & \text{if } i \in \mathfrak{e}_\ell \text{ and } i = \max\{j \,:\, j \in \mathfrak{e}_\ell\} \\ 0 & \text{if } i \notin \mathfrak{e}_\ell \end{cases} \tag{2.3}$$

and $W = \text{diag}\{w_1, \ldots, w_m\}$ where w_ℓ is the weight of edge $\mathfrak{e}_\ell$, i.e., $w_\ell = a_{ij}$ where $\mathfrak{e}_\ell = (i, j)$.

The Laplacian matrix is a linear diffusion operator associated with a graph. Likewise, we associate to each undirected graph a (possibly nonlinear) Laplacian operator $\mathcal{L} : \Omega^n \to \mathbb{R}^n$ defined as

$$\mathcal{L}(x) = BWf(B^T x) \tag{2.4}$$

where $f : \Omega \to \mathbb{R}$ is odd with $f(0) = 0$. We interpret f as acting element-wise on vector arguments to produce vector values, and we interpret B^T as a $\mathbb{Z}$-linear map from Ω^n to Ω^m. Note that the case $f = I$ and $\Omega = \mathbb{R}$ gives the Laplacian matrix, i.e., $\mathcal{L} = L$.

To analyze the estimators, it is useful to define the set of graph Laplacians which are balanced and satisfy a norm condition,

$$\mathbb{L}_n(\alpha) = \{L \in \mathbb{R}^{n\times n} : L\mathbf{1}_n = \mathbf{0}_n,\ \mathbf{1}_n^T L = \mathbf{0}_n^T,\ \|I_n - L - \Pi_n\|_2 \le \alpha\} \tag{2.5}$$

where $\alpha < 1$ and $\Pi_n = \mathbf{1}_n \mathbf{1}_n^T / n$. The following lemma characterizes this set of Laplacian matrices in the case when the graph is undirected.

Lemma 1. *Consider a connected and undirected graph with Laplacian matrix* $L \in \mathbb{R}^{n\times n}$. *Assume the nonzero Laplacian eigenvalues are in the interval* $[\lambda_{\min}, \lambda_{\max}]$. *Then* $k_p L \in \mathbb{L}_n(\alpha)$ *where*

$$\alpha = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}, \qquad k_p = \frac{2}{\lambda_{\max} + \lambda_{\min}}. \tag{2.6}$$

Proof. All undirected graphs are balanced, so $\mathbf{1}_n^T L = \mathbf{0}_n^T$. Since the graph is connected with nonzero eigenvalues in $[-\alpha, \alpha]$, the matrix $I - k_p L$ has a single eigenvalue at one with eigenvector $\mathbf{1}_n$, and the remaining eigenvalues are in $[-\alpha, \alpha]$. Then $I - k_p L - \Pi$ has the same eigenvalues, except the eigenvalue at one is shifted to zero. The Laplacian is symmetric since the graph is undirected, and the 2-norm is equal to the spectral radius for symmetric matrices, so $\|I - k_p L - \Pi\|_2 \leq \alpha$. Then we have $k_p L \in \mathbb{L}_n(\alpha)$. $\square$

For connected and undirected graphs, we let $\lambda_{\min}$ and $\lambda_{\max}$ with $0 < \lambda_{\min} \leq \lambda_{\max}$ denote lower and upper bounds on the nonzero eigenvalues of L. Then L has a simple eigenvalue at zero and all other eigenvalues are in the interval $[\lambda_{\min}, \lambda_{\max}]$.

To connect the dynamic average consensus problem with convex optimization, we can show that the Laplacian operator is the gradient of a scalar function.

Lemma 2. *If the graph is undirected and $f : \mathbb{T}^n \to \mathbb{R}^n$ is odd, then the Laplacian operator is the gradient of a scalar function. Specifically, we have $\mathcal{L}(x) = \nabla \mathbf{1}_m^T W \phi(B^T x)$ where $\nabla \phi = f$ and $\phi(0) = 0$.*

Proof. Since the graph is undirected, we can write the Laplacian operator as $\mathcal{L}(\cdot) = BWf(B^T \cdot)$. Similar to [**54**, Section IV.A], let ∂ denote the unit vector field pointing in the counterclockwise direction on $\mathbb{T}$. Since f is odd, the integral of the 1-form $f \cdot \langle \partial, \cdot \rangle$ around any smooth closed curve in $\mathbb{T}$ is zero. Thus, this 1-form is the differential of a smooth function $\phi : \mathbb{T} \to \mathbb{R}$, which is unique up to an additive constant (which we choose so that $\phi(0) = 0$). Therefore, $d/dt(\phi \circ x) \equiv f(x)\dot{x}$ for any curve $x : \mathbb{R} \to \mathbb{T}$. We then define $V : \mathbb{T}^m \to [0, \infty)$ as the sum

$$V(\theta) = \mathbf{1}_m^T W \phi(\theta) = \sum_{k=1}^{m} w_k \, \phi(\theta_k) \tag{2.7}$$

where $\theta = [\theta_1, \dots, \theta_m]^T \in \mathbb{T}^m$ and we interpret ϕ as acting element-wise on each element in the vector. Then the composition $V \circ B^T$ is

$$V(B^T x) = \sum_{(i,j)\in\mathcal{E}} a_{ij}\, \phi(x_i - x_j) \tag{2.8}$$

where $x = [x_1, \dots, x_n]^T \in \mathbb{T}^n$. Taking the derivative with respect to x_i gives

$$\frac{\partial}{\partial x_i} V(B^T x) = \sum_{j\in\mathcal{N}_{\text{in}}(i)} a_{ij}\, f(x_i - x_j) = [\mathcal{L}(x)]_i. \tag{2.9}$$

Therefore, $\mathcal{L}(x) = \nabla \mathbf{1}_m^T W \phi(B^T x)$ so the Laplacian is the gradient of a scalar function. □

2.2.1. Choosing the Edge Weights

The performance of estimators for dynamic average consensus depends on the edge weights used to form the Laplacian matrix. Specifically, the convergence rate is a monotonically non-increasing function of the ratio $\lambda_{\min}/\lambda_{\max}$ where $\lambda_{\min}$ and $\lambda_{\max}$ are bounds on the nonzero eigenvalues of the weighted Laplacian matrix (see Lemma 4). Therefore, the design of optimal estimators can be decomposed into two problems:

- choose the edge weights in a distributed manner to maximize $\lambda_{\min}/\lambda_{\max}$, and
- design the estimator to have the desired properties for any connected undirected weighted graph having Laplacian eigenvalues in the interval $[\lambda_{\min}, \lambda_{\max}]$.

If the graph topology is unknown, it is often useful to choose the weights to bound the eigenvalues of the Laplacian. For example, inverse out-degree weighting assigns

$$a_{ij} = \frac{1}{d_{\text{out}}(i) + d_{\text{out}}(j)}. \tag{2.10}$$

This decentralized weighting scheme restricts the eigenvalues of L to the region $D_0 \cap D_1$ where $D_x \subset \mathbb{C}$ denotes the closed unit disc centered at x [**15**]. In particular, we have $\rho(L) \leq 1$. This upper bound is tight in the sense that $\rho(L) = 1$ for some graphs using these weights (e.g., the graph having just two nodes connected by an edge). It also has the added advantage of producing symmetric (and therefore balanced) expected Laplacians under suitably symmetric packet-loss probability distributions (see [**15**] for details).

If the graph topology is known, more sophisticated methods can be used to design the weights. For undirected graphs, a semidefinite program can be solved to calculate the optimal weights which minimize α in Lemma 1 [**55**]. The case of directed graphs is studied in [**56**], but the optimization problem is non-convex since the spectral radius of $I - L - \Pi$ is minimized. Instead, we minimize the 2-norm, $\|I - L - \Pi\|_2$, which results in the convex optimization problem in Problem 2 for both directed and undirected graphs. Furthermore, we show in Chapter 5 this is the appropriate quantity to minimize when the graph is time-varying and balanced at each iteration.

Problem 2. *Given the node set $\mathcal{V}$ and the edge set $\mathcal{E}$, the Laplacian matrix in $\mathbb{L}_n(\alpha)$ which minimizes α and conforms to the graph is the solution to the following convex optimization problem:*

$$\begin{aligned} \min_{L \in \mathbb{R}^{n\times n}} \quad & \|I_n - L - \Pi_n\|_2 \\ \text{s.t.} \quad & \mathbf{0}_n = L\mathbf{1}_n \\ & \mathbf{0}_n^T = \mathbf{1}_n^T L \\ & 0 = L_{ij},\ (i,j) \notin \mathcal{E} \ \textit{and} \ i \neq j. \end{aligned} \tag{2.11}$$

2.3. Assumptions

The dynamic average consensus problem requires two types of assumptions; something must be known about both the signals and the communication graph. Information requires time to diffuse through the graph. If no assumptions are made on the signals, then the signals can change arbitrarily quickly causing the global average to change to any amount before the information has time to reach all the nodes. We now describe the types of assumptions we will make.

2.3.1. Input signals

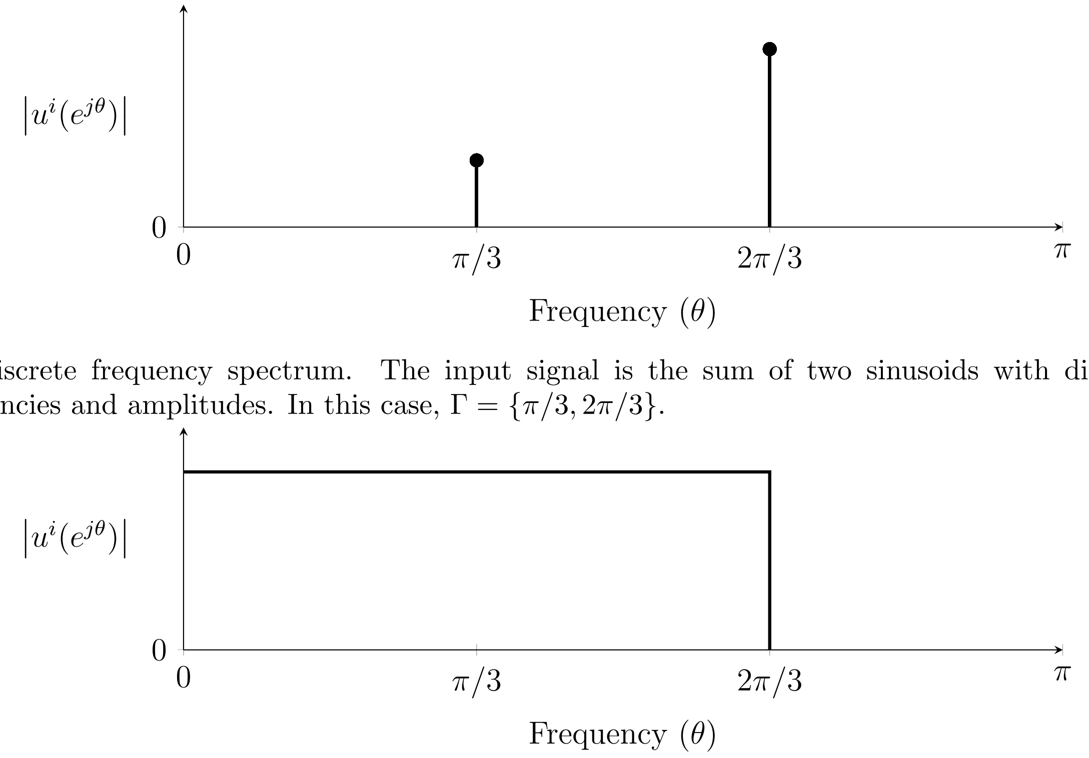
Consider the input signal to agent i given by u^i_k. Denote its z-transform as

$$u^i(z) = \sum_{k=-\infty}^{\infty} u^i_k \, z^{-k}. \tag{2.12}$$

The frequency spectrum of u^i is given by $u^i(z)$ evaluated on the unit circle in the complex plane, i.e., $u^i(e^{j\theta})$ for $\theta \in [0, 2\pi)$. Define the support of $u^i(e^{j\theta})$ as

$$\Gamma := \{\theta \in [0, 2\pi) : |u^i(e^{j\theta})| > 0\}. \tag{2.13}$$

We consider two classes of input signals. First, we design estimators for input signals whose frequency response has known discrete support, i.e., Γ is composed of a finite set of discrete frequencies. This class includes signals which are constant, polynomial, sinusoidal, and any linear combination of such signals. In general, this class includes signals with a known model, i.e., $u_i(z) = n_i(z)/d_i(z)$ where $n_i(z)$ and $d_i(z)$ are polynomials in z and $d_i(z)$ is known. If $d_i(z)$ is not known, it was shown in [**22**] that a frequency estimator can be used to estimate the frequency (i.e., the model $d_i(z)$) and the estimate used to design the estimator. For this class of signals, zero steady-state error can be achieved by putting a model of the input in the feedback loop of the estimator [**21**]. We design estimators for this class of signals in Chapter 4.



(a) Discrete frequency spectrum. The input signal is the sum of two sinusoids with different frequencies and amplitudes. In this case, $\Gamma = \{\pi/3, 2\pi/3\}$.

(b) Continuous frequency spectrum. The input signal is a bandlimited signal with cutoff frequency $2\pi/3$. In this case, $\Gamma = [0, 2\pi/3]$.

Figure 2.2. Frequency spectrum of example input signals. Plotted is the magnitude of the frequency response $|u_i(e^{j\theta})|$ as a function of the frequency θ.

The second class of signals that we consider includes signals with continuous support, i.e., Γ contains intervals of frequencies. This includes signals which are bandlimited, and we assume that the cutoff frequency is known. If the cutoff frequency is unknown, it could first be estimated and then the estimate used to design the estimator. For this class of signals, we first show that feedback designs as in the previous case are unsuitable to solve the dynamic average consensus problem. Instead, we propose a feedforward estimator design which can solve the problem with arbitrarily small (although nonzero) steady-state error. Estimators for this class of signals are designed in Chapter 5.

Examples of the two types of input signals are shown in Fig. 2.2. If the signals have both discrete and continuous frequencies in their spectrum, then estimators from the previous cases can be designed for each section of frequencies (either discrete or continous) and cascaded in series. This allows for a flexible design method based on whatever information is known about the signals.

2.3.2. Communication graph

Various assumptions can be made about the communication graph. One choice is whether the graph is assumed to be constant or time-varying. Time-varying graphs model many useful scenarios such as unreliable communication, asynchronous updates, as well as mobile agents capable of range-limited communication. Another consideration is whether communication is undirected or directed. Even in undirected networks, unreliable communication can cause communication links to become directed when packets are dropped. In the case of directed communication, a common assumption is that the graph is balanced [**15**]. This is equivalent to each agent having access to its out-degree, which is a restrictive assumption since this information is not available in general on directed graphs. However, it has been shown that the balanced assumption is a fundamental limitation of computing averages over a network. In fact, the dynamic average consensus problem is non-computable when (1) the communication is directed, (2) each node does not have (and make use of) global node identifiers, (3) the algorithm is deterministic, and (4) the communication is synchronous [**57**]. Therefore, we focus on the case of balanced graphs in this paper.

To summarize, common assumptions are that the communication graph

- is either constant or time-varying,
- is either balanced, undirected, or directed,
- is randomly generated from a given distribution,
- drops packets independently with a given probability distribution,
- has a known upper bound on the number of nodes, and/or
- has Laplacian matrix with nonzero eigenvalues contained in a known region.

2.4. Estimator Properties

We now define several properties of estimators for dynamic average consensus.

Definition 1 (Scalable)**.** *An estimator is said to be scalable if the total number of variables and computations in the system scales linearly with the number of agents.*

Definition 2 (Exact)**.** *Given a set of signals $\mathcal{U}$, an estimator is said to be exact when for any input signals $u \in \mathcal{U}$, the error of the estimator converges to zero when the internal states are initialized to zero.*

Definition 3 (Internally stable)**.** *An estimator is said to be internally stable when for any initial internal states and any bounded input signals, all internal states remain bounded in forward time.*

Definition 4 (Time-invariant)**.** *An estimator is said to be time-invariant when the estimator dynamics do not change with time.*

Definition 5 (p internal states variables)**.** *An estimator is said to use p internal state variables when each agent has p local scalar variables which it updates internally at each iteration.*

Definition 6 (q transmission variables)**.** *An estimator is said to use q transmission variables when each agent broadcasts q scalar variables in each packet.*

Recall that we define an iteration to be the computations that can be performed when each agent broadcasts no more than a single packet to its local neighbors. Therefore, we use the following definition for r-hop communication.

Definition 7 (r-hop communication)**.** *An estimator is said to use r-hop communication when each agent requires the past r packets of information from its neighbors in order to update the internal state variables and/or the output of the estimator.*

For an r-hop estimator, each agent must store the received packets of information from its neighbors for the past r iterations.

To implement an estimator with p internal state variables which uses r-hop communication with q transmission variables in each packet, agent i must have enough memory to store $p + q\,r\,|\mathcal{N}_{\text{in}}(i)|$ variables.

We now discuss robustness properties of estimators for dynamic average tracking. There are many forms of robustness to consider. For example, estimators may be robust to

(1) initial conditions,

(2) changes in the communication graph,

(3) noise in the input signals,

(4) delays in the communication, etc.

Robustness to initial conditions and robustness to changes in the communication graph are sometimes used interchangeably. When the graph changes, it can be viewed as restarting the estimator from a different set of initial conditions. If the estimator is robust to initial conditions and the graph is constant for a sufficient length of time after the change, then the estimator will converge to tracking the average of the inputs. For example, consider the simulation of the estimator which is not robust to changes in the graph in Fig. 2.3. The error converges to zero while the graph is constant, but then the graph is changed at iteration 300. The error immediately increases substantially, but then converges back to zero since the graph remains constant and the estimator is robust to initial conditions. After iteration 600, however, the graph changes at each iteration; the transient does not have time to decay, so the error remains large. For this reason, we refer to such estimators as robust to initial conditions, but not robust to changes in the graph.

We say that an estimator is robust to changes in the graph when the error remains the same whether the graph is changing at each iteration or is constant. An example is shown in Fig. 2.3. After the initial transient, the error remains small after both isolated and continuous changes in the graph.

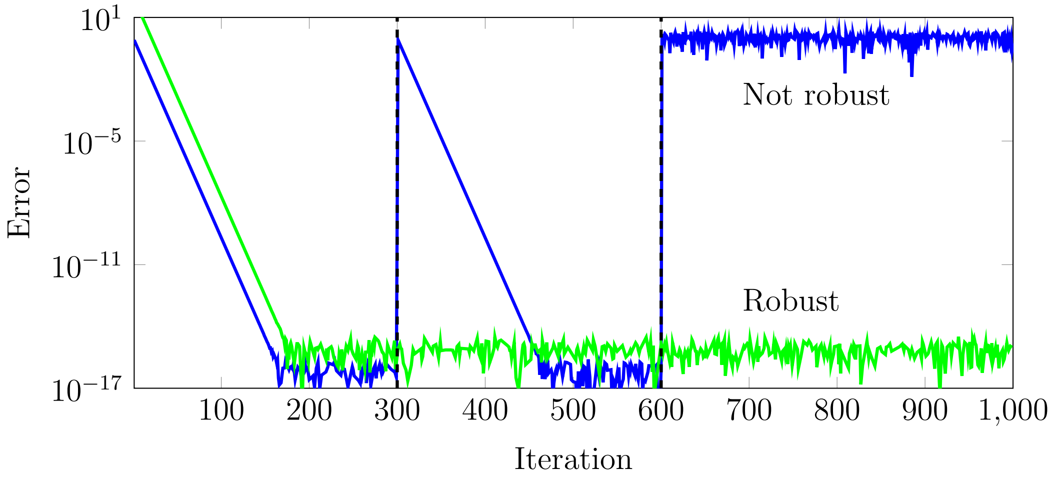


Figure 2.3. Simulation of estimators with different types of robustness properties. The estimators are (1) robust to initial conditions but not robust to changes in the graph (blue), and (2) robust to both initial conditions and changes in the graph (green). The graph changes at iteration 300 and then every iteration past 600.

The types of robustness described are now defined. First, we consider two ways in which an estimator can be robust to initial conditions.

Definition 8 (Locally convergent)**.** *An estimator is said to be locally convergent when there exists an open set of states such that the steady-state output is the same when the estimator states are initialized to any point in set.*

Definition 9 (Globally convergent)**.** *An estimator is said to be globally convergent when the steady-state of the output does not depend on the initial internal states.*

The steady-state value value does not depend on the initial conditions for estimators which are globally convergent. Therefore, we also use the term *robust to initial conditions* to describe estimators which are globally convergent (which is equivalent to being locally convergent for linear estimators).

Definition 10 (Ergodic)**.** *Suppose the input u is constant, and suppose the weighted communication graph changes at each time step according to some random process. Here we assume only that the expected graph is connected and balanced; the graph at each time step need not be connected nor balanced. Let L_k denote the resulting Laplacian at time k, and assume L_k is i.i.d. and independent of the input and any initial states. Then we call an estimator ergodic when for any sufficiently small nonzero variance of the Laplacian process L_k, the output process y_k is asymptotically ergodic in the mean, i.e., its time average converges to its statistical average as $k \to \infty$.*

Definition 11 (Robust to changes in the graph)**.** *Given a set of graphs $\mathbb{G}$, consider implementing an estimator in the following two scenarios:*

(1) The graph $\mathcal{G} \in \mathbb{G}$ is fixed.

(2) The graph at time k is $\mathcal{G}_k \in \mathbb{G}$ for $k = 1, 2, \ldots$.

An estimator is said to be robust to changes in the graph *when the steady-state error using any sequence of graphs in (2) is no greater than the steady-state error using the worst-case constant graph in (1).*

For one-hop estimators in which agents communicate only with their one-hop neighbors at each time step, the statistical average of the output y_k is the output of the estimator when the switching Laplacian L_k is replaced by its expected value. For multi-hop estimators in which higher powers of the Laplacian appear, the statistical average of the output y_k is the output of the estimator when each power of L_k is replaced by its expected value. If an ergodic estimator is exact under these expected Laplacian powers, then under the switching Laplacian a local low-pass filter can be applied to each local output to obtain the exact global average. Conditions for an estimator to be ergodic are given in 3.

These properties of the estimator may depend on the graph; for example, typically an estimator can be exact only for connected graphs, and typically an estimator can be internally stable only for graphs whose Laplacian eigenvalues satisfy a known upper bound.

2.5. Representation of Estimators

We now consider how to represent estimators for dynamic average consensus. A single type of representation (e.g., the state-space form) could always be used to describe the estimators. However, it is often useful to consider different representations since these can lead to various insights into the algorithm. The two main types of representations that we consider are the state-space representation (in the time-domain) which is useful for *analysis*, and the block diagram (in the frequency-domain) which is useful for *design*.

Throughout the paper, we consider *diffusion algorithms* which are characterized by each agent taking weighted averages among its neighbors. We use the Laplacian operator to implement diffusion algorithms on graphs. Consider applying the Laplacian operator to a vector $x \in \mathbb{R}^n$. This is implemented on agent i by taking a weighted average of the difference between neighbors passed through the function f,

$$[\mathcal{L}(x)]_i = \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij} f(x_i - x_j). \tag{2.14}$$

This operation only requires information from the in-neighbors of agent i and is therefore able to be implemented in a distributed manner.

The first estimator representation we consider is the polynomial linear protocol [**15**] which uses the Laplacian matrix for diffusion.

Definition 12. *An r-hop polynomial linear protocol (PLP) of dimension p is the collection $\Sigma(L) = [A(L), B(L), C(L), D(L)]$ where*

$$\begin{aligned} A(L) &= \sum_{i=0}^{r} A_i \otimes L^i \qquad & B(L) &= \sum_{i=0}^{r} B_i \otimes L^i \\ C(L) &= \sum_{i=0}^{r} C_i \otimes L^i \qquad & D(L) &= \sum_{i=0}^{r} D_i \otimes L^i \end{aligned} \tag{2.15}$$

are polynomials in the Laplacian matrix L which describe the linear system

$$\begin{aligned} x_{k+1} &= A(L)x_k + B(L)u_k, \quad x_0 \in \mathbb{R}^{np} \\ y_k &= C(L)x_k + D(L)u_k. \end{aligned} \tag{2.16}$$

The sizes of matrices and vectors are $L \in \mathbb{R}^{n \times n}$, $A_i \in \mathbb{R}^{p \times p}$, $B_i \in \mathbb{R}^p$, $C_i^T \in \mathbb{R}^p$, $D_i \in \mathbb{R}$, $u_k \in \mathbb{R}^n$, $x_k \in \mathbb{R}^{np}$, and $y_k \in \mathbb{R}^n$.

In all of the block diagrams, a line indicates a signal composed of q variables per agent for some q. That is, the signal can be decomposed as $x = [x_1^T, \ldots, x_q^T]^T \in \mathbb{R}^{nq}$ where the i^{th} component of x_j is associated with agent i. In order for the estimator to be distributed, the block diagram may contain two types of blocks:

Diagonal blocks: The element on the i^{th} diagonal is implemented on agent i and no communication is required.

Diffusion blocks: These blocks must have sparsity structure corresponding to the edges in the graph and require communication among neighboring agents. We use the graph Laplacian for diffusion.

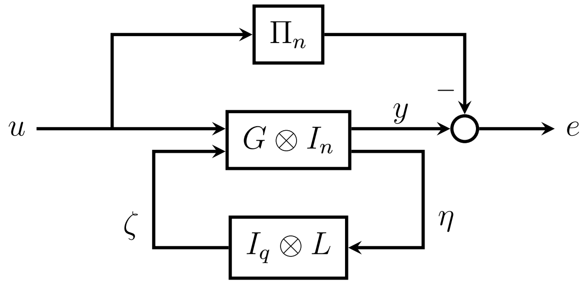


Figure 2.4. Block diagram of a general one-hop estimator.

We now give a block diagram representation of a one-hop polynomial linear protocol estimator in Fig. 2.4. This representation is useful in that it allows us to determine exactly how many variables must be transmitted in each packet. Since q variables are multiplied by the Laplacian matrix, each agent must broadcast a single packet containing q variables to neighboring agents at each iteration. The state-space representation is given by

$$
\begin{aligned}
x_{k+1} &= (A \otimes I_n)x_k + (B_1 \otimes I_n)u_k + (B_2 \otimes I_n)\zeta_k, \qquad x_0 \in \mathbb{R}^{np} \\
y_k &= (C_1 \otimes I_n)x_k + (D_{11} \otimes I_n)u_k + (D_{12} \otimes I_n)\zeta_k \\
\eta_k &= (C_2 \otimes I_n)x_k + (D_{21} \otimes I_n)u_k \\
\zeta_k &= (I_q \otimes L_k)\eta_k \\
e_k &= y_k - \Pi_n u_k
\end{aligned}
\tag{2.17}
$$

where $A \in \mathbb{R}^{p\times p}$, $B_1 \in \mathbb{R}^p$, $B_2 \in \mathbb{R}^{p\times q}$, $C_1 \in \mathbb{R}^{1\times p}$, $C_2 \in \mathbb{R}^{q\times p}$, $D_{11} \in \mathbb{R}$, $D_{12} \in \mathbb{R}^{1\times q}$, and $D_{21} \in \mathbb{R}^q$. We set $D_{22} = \mathbf{0}_{q\times q}$ so that the system is well-posed. This is a polynomial linear protocol where the state-space matrices in (2.15) are given by

$$
\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \left[\begin{array}{c|c} A & B_1 \\ \hline C_1 & D_{11} \end{array}\right] \otimes I_n + \left[\begin{array}{c|c} B_2 C_2 & B_2 D_{21} \\ \hline D_{12} C_2 & D_{12} D_{21} \end{array}\right] \otimes L. \tag{2.18}
$$

Another useful block diagram with feedback structure is shown in Fig. 2.5. In the block diagram, each signal is n-dimensional with the i^{th} component associated with agent i. In order for the estimator to use one-hop communication, there must be a strictly proper transfer function on the path between any two connected Laplacian blocks. In this case each agent can broadcast a single packet at each iteration, and the number of variables in the packet corresponds to the number of Laplacian blocks in the diagram. For example, the estimator in Fig. 2.5 requires three variables to be transmitted in each packet, although the ellipses indicate how to generalize the diagram for a general number of transmission variables. This representation is useful because many properties of the estimator can be identified simply from the structure of the block diagram, as we will show shortly.

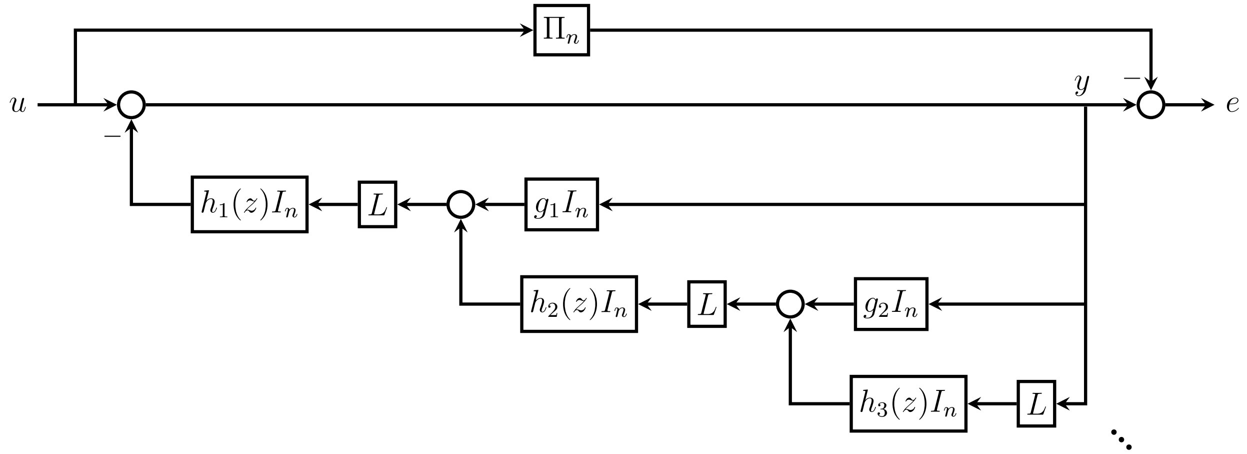


Figure 2.5. Block diagram of a general feedback estimator.

Bai et al. [**21**] give the block diagram for the generalized PI estimator in which each agent has a general number of internal state variables and two variables are communicated with neighboring agents at each iteration. The estimator in Fig. 2.5 generalizes this to allow for a general number of variables to be transmitted to neighboring agents at each iteration.

Each of the estimator representations have been presented using the Laplacian matrix for diffusion. These can all be trivially generalized to use the more general Laplacian operator.

2.6. Separated System

Instead of analyzing the entire system at once, we can decompose estimators into smaller systems and analyze the resulting systems separately. First, we show how to separate a polynomial linear protocol according to the eigenvalues of the Laplacian matrix when the graph is constant.

Theorem 1 (Separated system (PLP))**.** *Let $\mathcal{G}$ be a constant graph with Laplacian matrix L. Let $L = V\Lambda V^{-1}$ where $V = [v_1, \ldots, v_n]$, $V^{-1} = [\bar{v}_1, \ldots, \bar{v}_n]^T$, and $\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_n)$. Consider a polynomial linear protocol $\Sigma(L)$ with initial condition x_0. Define the i^{th} separated* *system as*

$$
\begin{aligned}
\bar{x}^i_{k+1} &= A(\lambda_i)\bar{x}^i_k + B(\lambda_i)(\bar{v}_i^T u_k), \quad \bar{x}^i_0 = (\bar{v}_i^T \otimes I_p)x_0 \in \mathbb{R}^p \\
\bar{y}^i_k &= C(\lambda_i)\bar{x}^i_k + D(\lambda_i)(\bar{v}_i^T u_k)
\end{aligned}
\tag{2.19}
$$

for $i = 1, \ldots, n$. Then the output of $\Sigma(L)$ is

$$
y_k = \sum_{i=1}^{n} v_i \, \bar{y}^i_k. \tag{2.20}
$$

Proof. Apply the change of variable $\bar{x}_k = (V^{-1} \otimes I_p)x_k$ to the estimator in (2.16). This gives the equivalent diagonal system,

$$
\begin{aligned}
\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] &\sim \left[\begin{array}{c|c} (V^{-1} \otimes I_p)A(L)(V \otimes I_p) & (V^{-1} \otimes I_p)B(L) \\ \hline C(L)(V \otimes I_p) & D(L) \end{array}\right] \\
&= \left[\begin{array}{c|c} A(\Lambda) & B(\Lambda)V^{-1} \\ \hline VC(\Lambda) & VD(\Lambda)V^{-1} \end{array}\right] \\
&= \left[\begin{array}{ccc|c} A(\lambda_1) & & & B(\lambda_1)\bar{v}_1^T \\ & \ddots & & \vdots \\ & & A(\lambda_n) & B(\lambda_n)\bar{v}_n^T \\ \hline v_1 C(\lambda_1) & \cdots & v_n C(\lambda_n) & \sum_{i=1}^n (v_i \bar{v}_i^T) D(\lambda_i) \end{array}\right]
\end{aligned}
$$

whose output is given by (2.20). □

Theorem 1 separates the system according to the eigenvalues of the Laplacian matrix when the graph is constant. For any $\lambda \in \operatorname{eig}(L)$ with left and right eigenvectors $\bar{v}$ and v, respectively, the separated system is constructed as follows:

- Replace the input with $\bar{v}^T u$.

- Replace the Laplacian matrix with λ.
- Multiply the output by v.

Then the output of the estimator is the sum over all eigenvalues of L of each of the resulting outputs. We refer to the separated system corresponding to the eigenvalue $\lambda_1 = 0$ as the *consensus system* and to λ_i, $i = 2, \ldots, n$ as the *disagreement systems*.

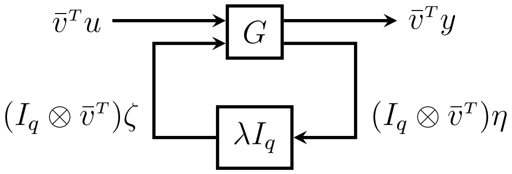


Figure 2.6. Separated system of the general one-hop estimator in Fig. 2.4.

Since the estimators in Figs. 2.4 and 2.5 are polynomial linear protocols, they can also be separated using this procedure. In that case, the block diagram of the separated system for each estimator is shown in Figs. 2.6 and 2.7, respectively.

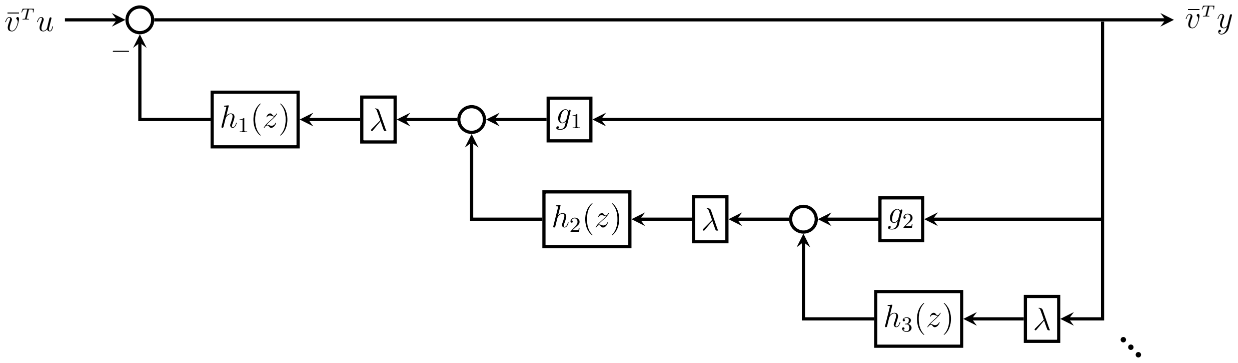


Figure 2.7. Separated system of the feedback estimator in Fig. 2.5.

In order to separate the system using Theorem 1, the graph must be constant and the Laplacian matrix must be used for diffusion. We can still separate the system when the graph is time-varying and/or the Laplacian operator is used, but then the system can only be separated into the *consensus system* and *disagreement system*.

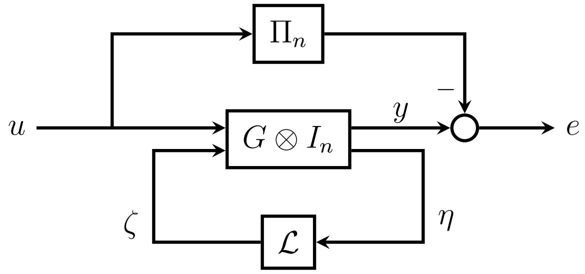


Figure 2.8. Block diagram of a one-hop estimator which transmits one variable per iteration and uses the Laplacian operator.

Consider the estimator in Fig. 2.8. This is simply the estimator in Fig. 2.4 with $q = 1$ and the Laplacian matrix L replaced with the Laplacian operator $\mathcal{L}$. To separate the system, we first define the reduced Laplacian.

Definition 13 (Reduced Laplacian)**.** *Let $v_1 = \mathbf{1}_n/\sqrt{n}$ and $Q \in \mathbb{R}^{n\times(n-1)}$ be such that the matrix $[v_1\ Q]$ is orthogonal. Then the* reduced Laplacian operator *is defined as $\tilde{\mathcal{L}} : \Omega^{n-1} \to \mathbb{R}^{n-1}$ where*

$$\tilde{\mathcal{L}}(x) = Q^{\mathsf{T}} BWf(B^{\mathsf{T}} Qx). \tag{2.21}$$

We now separate the system into the consensus system and the disagreement system.

Theorem 2 (Separated system)**.** *Let $\mathcal{G}$ be a graph which is connected and undirected at each iteration with Laplacian operator $\mathcal{L}_k$ at time k. Let $Q \in \mathbb{R}^{n\times(n-1)}$ be such that $V := [v_1\ Q]$ is orthogonal where $v_1 = \mathbf{1}_n/\sqrt{n}$. Consider the estimator in Fig. 2.8 with initial* *condition x_0. Define the consensus system as*

$$
\begin{aligned}
x^c_{k+1} &= (A \otimes I_n) x^c_k + (B_1 \otimes I_n) u^c_k, \qquad x^c_0 = (I_p \otimes v_1^T) x_0 \\
y^c_k &= (C_1 \otimes I_n) x^c_k + (D_{11} \otimes I_n) u^c_k \\
e^c_k &= y^c_k - u^c_k.
\end{aligned} \tag{2.22}
$$

where $u^c_k = v_1^T u_k$, and the disagreement system as

$$
\begin{aligned}
x^d_{k+1} &= (A \otimes I_n) x^d_k + (B_1 \otimes I_n) u^d_k + (B_2 \otimes I_n) \zeta^d_k, \qquad x^d_0 = (I_p \otimes Q^T) x_0 \\
y^d_k &= (C_1 \otimes I_n) x^d_k + (D_{11} \otimes I_n) u^d_k + (D_{12} \otimes I_n) \zeta^d_k \\
\eta^d_k &= (C_2 \otimes I_n) x^d_k + (D_{21} \otimes I_n) u^d_k + (D_{22} \otimes I_n) \zeta^d_k \\
\zeta^d_k &= \tilde{\mathcal{L}}_k(\eta^d_k) \\
e^d_k &= y^d_k
\end{aligned} \tag{2.23}
$$

where $u^d_k = Q^T u_k$. Then the error of the estimator in Fig. 2.8 is

$$
e_k = v_1 e^c_k + Q e^d_k. \tag{2.24}
$$

Proof. Since the graph is undirected at each iteration, we have $L_k v_1 = L_k^T v_1 = \mathbf{0}_n$ for all $k \geq 0$ where $v_1 = \mathbf{1}_n/\sqrt{n}$. Let $Q \in \mathbb{R}^{n \times (n-1)}$ be such that $V := [v_1 \; Q]$ is orthogonal. Then $I_n = VV^T = v_1 v_1^T + QQ^T$. Define the consensus and disagreement variables as

$$
\begin{aligned}
&x^c_k = (I_p \otimes v_1^T) x_k \quad && y^c_k = v_1^T y_k \quad && \eta^c_k = v_1^T \eta_k \quad && \zeta^c_k = v_1^T \zeta_k \quad && e^c_k = v_1^T e_k \quad && u^c_k = v_1^T u_k \\
&x^d_k = (I_p \otimes Q^T) x_k \quad && y^d_k = Q^T y_k \quad && \eta^d_k = Q^T \eta_k \quad && \zeta^d_k = Q^T \zeta_k \quad && e^d_k = Q^T e_k \quad && u^d_k = Q^T u_k,
\end{aligned}
$$

respectively. Since $\mathcal{L}(\cdot) = BWf(B^T\cdot)$ with $B^T v_1 = \mathbf{0}_m$, we have

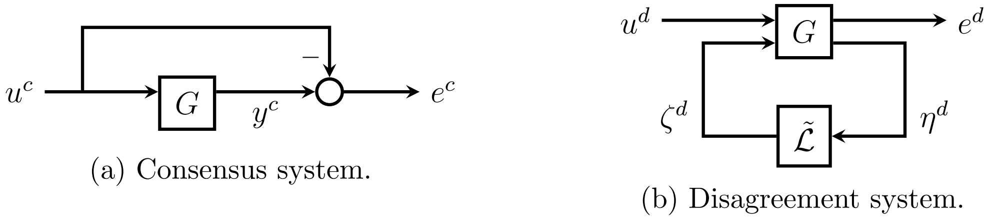
$$\zeta_k^c = v_1^T \mathcal{L}_k(\eta_k) = 0$$

for all $k \geq 0$, and

$$\zeta_k^d = Q^T \mathcal{L}_k((v_1 v_1^T + QQ^T)\eta_k) = Q^T \mathcal{L}_k(QQ^T \eta_k) = \tilde{\mathcal{L}}_k(\eta_k^d).$$

Writing the system in Fig. 2.8 in terms of the consensus and disagreement variables produces the consensus system in (2.22) and the disgreement system in (2.23) where the output is $e_k = (v_1 v_1^T + QQ^T)e_k = v_1 e_k^c + Q e_k^d$. □

For the estimator in Fig. 2.8, the block diagrams of the consensus and disagreement systems are shown in Fig. 2.9.



(a) Consensus system.

(b) Disagreement system.

Figure 2.9. Separated system of the one-hop estimator in Fig. 2.8 which uses the Laplacian operator.

The separated system gives insight into the design of dynamic average tracking estimators. When the graph is both connected and balanced, the Laplacian matrix has a single eigenvalue at zero and the corresponding left and right eigenvectors are both $\mathbf{1}_n/\sqrt{n}$. In that case, if the consensus system has unity gain and all the disagreement systems have zero gain, then $y_k = \Pi_n u_k$ so the estimator solves the dynamic average consensus problem. Therefore, we want to design the system to have unity gain in the consensus direction and zero gain in the disagreement directions.

2.7. Analysis of Estimators

We can now characterize properties of the estimators in Figs. 2.4 and 2.5 by analyzing their separated systems. We focus on the case when the input signals are constant, the linear Laplacian matrix is used, and the graph is constant, connected, and undirected. The case for time-varying inputs with a known internal model is similar where the integrator is replaced by the model of the input; see [**21**] for details.

Theorem 3 (Estimator properties, Fig. 2.4)**.** *Let $\mathcal{G}$ be a constant, connected, and balanced graph with Laplacian matrix L, and let the input signals be given by $u(z) = n(z)/d(z)$ where $n(z)$ and $d(z)$ are polynomials in z. Then the estimator in Fig. 2.4 has the following properties:*

Robust to initial conditions $\iff$ $\lim\limits_{k\to\infty} C(\lambda)A(\lambda)^k = \mathbf{0}_{1\times p}$ *for all* $\lambda \in \mathrm{eig}(L)$

Internally stable $\iff$ $(A(\lambda), B(\lambda))$ *is bounded-input, bounded state stable for all* $\lambda \in \mathrm{eig}(L)$

Exact $\iff$ $H(z,\lambda) = \begin{cases} 1, & \lambda = 0 \\ 0, & \lambda \in \mathrm{eig}(L) \setminus \{0\} \end{cases}$ *for all* z *s.t.* $d(z) = 0$

Ergodic $\Longleftarrow$ $D_{12}C_2 = \mathbf{0}_{1\times p}$ *and* $D_{12}D_{21} = 0$ *and robust to ICs* *where $H(z,\lambda) := C(\lambda)[zI - A(\lambda)]^{\dagger}B(\lambda) + D(\lambda)$ with $A(\lambda)$, $B(\lambda)$, $C(\lambda)$, and $D(\lambda)$ defined in* (2.18)*. Furthermore, the estimator has convergence rate*

$$\rho = \max_{\lambda \in \mathrm{eig}(L)} \rho\left(\tilde{A}(\lambda)\right) \tag{2.25}$$

where $(\tilde{A}(\lambda), \tilde{B}(\lambda), \tilde{C}(\lambda), \tilde{D}(\lambda))$ is an observable decomposition of the system.

Proof. Since the graph is constant, we can separate the system according to the eigenvalues of L using Theorem 1. Then transfer function of each subsystem from input $v_i^T u$ to output y^i is given by (2.18) with L replaced by λ_i. The output of the subsystem at time k is

$$y_k^i = C(\lambda)A(\lambda)^k(I_p \otimes \bar{v}_i^T)x_0 + C(\lambda)\sum_{j=0}^{k-1} A(\lambda)^{k-j-1}B(\lambda)u_j + D(\lambda)u_k. \tag{2.26}$$

The steady-state output does not depend on the initial conditions if and only if $\mathbf{0}_{1\times p} = \lim_{k\to\infty} C(\lambda)A(\lambda)^k$ for each subsystem. Also, the complete system is bounded-input, bounded-state stable if and only if each subsystem is bounded-input, bounded-state stable. The condition for the estimator to be exact is from [**15**, Thm. 5]. In Chapter 3 we show that a sufficient condition for ergodicity is that the system is robust to initial conditions and the output is independent of the Laplacian matrix, which is the case if $D_{12}C_2 = \mathbf{0}_{1\times p}$ and $D_{12}D_{21} = 0$. The convergence rate is then the maximum convergence rate of each of the subsystems. Since the subsystem may contain unobservable states, we use an observable decomposition to only take into account the states which affect the output. □

Theorem 3 implies that an estimator must contain an internal model of the input signals in order to be exact, and the internal model must be uncontrollable and unobservable for the estimator to be internally stable and robust to initial conditions, respectively (see [**15**] for details). These conditions could be used to design estimators, although they do not lead to a simple design procedure. Instead, we consider the feedback structure in Fig. 2.5. The properties of this estimator can easily be determined from the structure of the block diagram. Before we restate Theorem 3 for the estimator in Fig. 2.5, we need the following lemma.

Lemma 3. *The transfer function of the separated system in Fig. 2.7 is*

$$\frac{\bar{v}^T y(z)}{\bar{v}^T u(z)} = \frac{1}{F(z,\lambda)} \tag{2.27}$$

where

$$F(z,\lambda_i) = 1 + \sum_{\ell=1}^{q} f_\ell(z)\lambda_i^\ell \qquad \textit{and} \qquad f_\ell(z) = g_\ell \prod_{j=1}^{\ell} h_j(z), \quad \ell = 1,\ldots,q \tag{2.28}$$

with $g_q = 1$ where q is the number of Laplacian blocks.

We make the following assumption which excludes algebraic loops in Fig. 2.5.

Assumption 1. *The transfer functions $f_\ell(z)$ for $\ell = 1,\ldots,q$ are either strictly proper or identically zero.*

We can now restate Theorem 3 for the estimator in Fig. 2.5. This characterizes the properties of the estimator based on the structure of the block diagram.

Theorem 4 (Estimator properties, Fig. 2.5)**.** *Let $\mathcal{G}$ be a constant, connected, and balanced graph with Laplacian matrix L, and let the input signals be given by $u(z) = n(z)/d(z)$ where $n(z)$ and $d(z)$ are polynomials in z. Assume that transfer function* (2.27) *has all poles strictly inside the unit circle for all $\lambda \in eig(L)\backslash\{0\}$. Then the estimator in Fig. 2.5 has the* *following properties:*

$$\begin{aligned}
&\textit{Robust to initial conditions} && \iff && h_1(z) \textit{ is stable}\\
&\textit{Internally stable} && \iff && \textit{for all } \ell = 1, \ldots q,\ h_\ell(z) \textit{ has no poles strictly outside the}\\
& && && \quad \textit{unit circle and no repeated poles on the unit circle}\\
&\textit{Exact} && \iff && \textit{for each } z_\star \textit{ such that } 0 = d(z_\star),\\
& && && \quad h_\ell(z) \textit{ has a pole at } z_\star \textit{ for some } \ell = 1, \ldots, q\\
&\textit{Ergodic} && \Longleftarrow && h_1(z) \textit{ is strictly proper, and robust to ICs.}
\end{aligned}$$

Furthermore, the estimator has convergence rate

$$\rho = \max\left\{\rho_{\text{cons}}, \rho_{\text{dis}}\right\} \tag{2.29}$$

where

$$\rho_{\text{cons}} = \max_{z\in\mathbb{C}} |z| \quad \text{s.t. } 0 = d_1(z) \tag{2.30}$$

$$\rho_{\text{dis}} = \max_{\substack{z\in\mathbb{C}\\ \lambda\in\text{eig}(L)}} |z| \quad \text{s.t. } 0 = F(z,\lambda) \tag{2.31}$$

with $h_1(z) = n_1(z)/d_1(z)$.

Proof. Since the graph is balanced and connected, the Laplacian has a single eigenvalue at zero and all other eigenvalues are positive, so we can order them as $0 = \lambda_1 < \lambda_2 \leq \lambda_3 \leq \ldots \leq \lambda_n$ with corresponding eigenvectors v_i where $v_1 = \mathbf{1}_n/\sqrt{n}$. The output of the full system is given by equation (2.20), so we can analyze the separated system for each λ_i, $i = 1, \ldots, n$.

1) Applying the final value theorem to the output of the separated system with input $u(z) = n(z)/d(z)$,

$$\lim_{k\to\infty} v_i^T y_k = \lim_{z\to 1}(z-1)v_i^T y(z) = \lim_{z\to 1} v_i^T \frac{n(z)}{d(z)} \frac{z-1}{1+\sum_{\ell=1}^{q} f_\ell(z)\lambda_i^\ell}$$

if the limit exists. For $i = 2,\ldots,n$, the limit on the right exists if and only if $f_\ell(z)$ contains a pole at each root of $d(z)$ for some ℓ so that the system multiplied by the input has all poles strictly inside the unit circle. Then the output only depends on the $\lambda_1 = 0$ system. From the block diagram, it is clear that $v_1^T y_k = v_1^T u_k$ for all k (when the states of $h_1(z)$ are initialized to zero). Therefore, the output of the full system from equation (2.20) is $\lim_{k\to\infty} y_k = v_1 v_1^T u$ which is the exact average of the inputs.

2) For $i = 2,\ldots,n$, the separated system is internally stable since the transfer function (2.27) has all poles strictly inside the unit circle. For $i = 1$, the input to $h_\ell(z)$ for $\ell = 1,\ldots,q$ is zero since it is first multiplied by $\lambda_1 = 0$. The output of $h_\ell(z)$ is guaranteed to be bounded if and only if $h_\ell(z)$ has no poles strictly outside the unit circle and no repeated poles on the unit circle for all $\ell = 1,\ldots,q$.

3) For $i = 2,\ldots,n$, the separated system is stable so the output due to the initial conditions is zero. For $i = 1$, the output due to the initial conditions of $h_1(z)$ is zero if and only if $h_1(z)$ is stable.

4) From Thm. 9 in Chapter 3, an estimator is ergodic if it achieves consensus for the expected Laplacian, the consensus system does not have an observable eigenvalue at $z = 1$, and each local output y_k^i at time k depends only on the local input u_k^i and the local internal state at time k, not on any information collected from neighbors. The output of

each estimator only depends on the internal state at time k if and only if $h_1(z)$ is strictly proper. □

Theorem 4 shows that the properties of the estimator depend on how the model of the input appears in the block diagram. Specifically, we have the following properties as shown in Figure 2.10:

- To be exact, the estimator must contain a model of the input in the feedback path.
- To be internally stable, the output must pass through the Laplacian before reaching any marginally stable transfer functions.
- To be robust to initial conditions, the output of any marginally stable transfer function must pass through the Laplacian before reaching the output.
- To be ergodic, it is sufficient that the estimator is robust to initial conditions and any state passed through the Laplacian is filtered by a strictly proper transfer function before reaching the output.

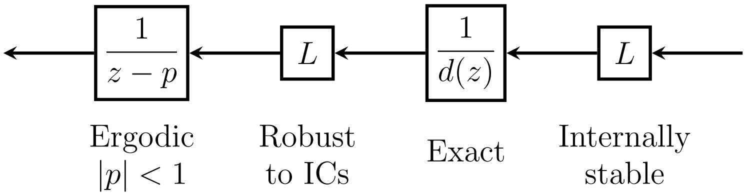


Figure 2.10. Properties of a feedback estimator based on the structure of the block diagram.

An important parameter in the design of estimators is the ratio $\lambda_r := \lambda_{\min}/\lambda_{\max}$ where $\lambda_{\min}$ and $\lambda_{\max}$ are lower and upper bounds, respectively, on the nonzero eigenvalues of the Laplacian matrix (in the case of undirected graphs). This can be interpreted as the inverse condition number of the reduced Laplacian matrix, i.e., the Laplacian matrix in directions orthogonal to $\mathbf{1}_n$. We now show that the convergence rate in (2.29) is a monotonically non-increasing function of λ_r.

Lemma 4. *The convergence rate ρ is a monotonically non-increasing function of the ratio $\lambda_{\min}/\lambda_{\max}$.*

Proof. Instead of designing the estimator for $\lambda \in [\lambda_{\min}, \lambda_{\max}]$, we can do the design for $\lambda \in [\lambda_r, 1]$ where $\lambda_r = \lambda_{\min}/\lambda_{\max}$ and then scale $h_\ell(z)$ by $\lambda_{\max}$ in Figure 2.5. Define $\rho(\lambda)$ to be the worst-case asymptotic convergence rate over $[\lambda, 1]$. Then for any $\lambda > \lambda_r$, the interval $[\lambda, 1]$ is a subset of $[\lambda_r, 1]$. Therefore, $\rho(\lambda)$ cannot be greater than $\rho(\lambda_r)$, so ρ is a monotonically non-increasing function of $\lambda_{\min}/\lambda_{\max}$. □

2.8. Palindromic Transformation

In this section, we develop a general way of accelerating the convergence rate of estimators when the graph is undirected. Suppose we have an estimator which converges linearly with rate 2ρ. Furthermore, suppose the closed-loop poles are all contained in the real interval $[-2\rho, 2\rho]$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ where $\lambda_{\min}$ and $\lambda_{\max}$ are lower and upper bounds on the nonzero Laplacian eigenvalues, respectively. Then we can perform a transformation which maps the interval $[-2\rho, 2\rho]$ to the circle $\rho\mathbb{T}$, so the resulting estimator has convergence rate ρ. To do this mapping, we use palindromic polynomials which are defined below.

Definition 14 (ρ-palindromic polynomial)**.** *A polynomial p of degree at most $2d$ is called* ρ-palindromic *if and only if there exists a polynomial q of degree at most d such that*

$$p(z) = z^d\, q(z + \rho^2/z). \tag{2.32}$$

Remark 1. *The function $z + \rho^2/z$ maps $\rho\mathbb{T}$ to the interval $[-2\rho, 2\rho]$.*

Definition 15 (ρ-palindromic rational function)**.** *A rational function P is called ρ-*palindromic *if and only if there exists a rational function Q such that $P(z) = Q(z + \rho^2/z)$.*

Table 2.1. Examples of ρ-palindromic rational function pairs.

$P(z)$	$Q(z)$
$\dfrac{z}{z^2 - \beta z + \rho^2}$	$\dfrac{1}{z - \beta}$
$\dfrac{z}{(z-1)(z-\rho^2)}$	$\dfrac{1}{z - (1 + \rho^2)}$
$\dfrac{z(z^2 - \gamma z + \rho^2)}{(z^2 - \beta_1 z + \rho^2)(z^2 - \beta_2 z + \rho^2)}$	$\dfrac{z - \gamma}{(z - \beta_1)(z - \beta_2)}$
$\dfrac{z(z^2 - \gamma z + \rho^2)}{(z^2 - 2\cos(\theta)z + 1)(z^2 - 2\rho^2\cos(\theta)z + \rho^4)}$	$\dfrac{z - \gamma}{z^2 - 2(1 + \rho^2)\cos(\theta)z + (1 + 2\rho^2\cos(2\theta) + \rho^4)}$

Lemma 5. *Suppose the rational function F is ρ-palindromic and that the zeros of F are all on $\rho\mathbb{T}$. Then there exists a rational function $\tilde{F}$ such that $F(z) = \tilde{F}(z + \rho^2/z)$ and the zeros of $\tilde{F}$ are all real and contained in the interval* $[-2\rho, 2\rho]$.

Proof. Since F is ρ-palindromic, there exists a rational function $\tilde{F}$ such that $F(z) = \tilde{F}(z + \rho^2/z)$. Since the zeros of F are on $\rho\mathbb{T}$, each zero has the form $z = \rho\, e^{j\theta}$ for some $\theta \in [-\pi, \pi]$ where $j = \sqrt{-1}$. The zeros of F are the solutions to

$$0 = F(z) = \tilde{F}(z + \rho^2/z) = \tilde{F}\Big(\rho\, e^{j\theta} + \frac{\rho}{e^{j\theta}}\Big) = \tilde{F}(2\rho\cos\theta)$$

for some $\theta \in [-\pi, \pi]$, so the zeros of $\tilde{F}$ are real and contained in the interval $[-2\rho, 2\rho]$. □

Lemma 5 shows that if the roots of F are on $\rho\mathbb{T}$ and F is ρ-palindromic, then there is a function $\tilde{F}$ of *less degree* whose roots are in the real interval $[-2\rho, 2\rho]$. Each real root of $\tilde{F}$ corresponds to two complex conjugate roots of F, so $\tilde{F}$ is half the degree in z as F. If $0 = \tilde{F}(z,\lambda)$ is the characteristic equation for an estimator which has all of its poles in $[-2\rho, 2\rho]$, then the estimator with characteristic equation $0 = F(z,\lambda) = \tilde{F}(z + \rho^2/z, \lambda)$ has all of its poles on $\rho\mathbb{T}$. This is summarized in the following corollary.

Theorem 5. *Let $\mathcal{G}$ be a constant, undirected graph with nonzero Laplacian eigenvalues in the interval $[\lambda_{\min}, \lambda_{\max}]$. Consider an estimator with characteristic equation $0 = F(z, L)$. Suppose $0 = F(1+\rho^2, \lambda)$ and all solutions of $0 = F(z,\lambda)$ are in the real interval $[-2\rho, 2\rho]$ for every $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. Then the estimator with characteristic equation $0 = F(z + \rho^2/z, L)$ is exact for constant inputs and converges linearly with rate ρ.*

Proof. First, note that $F(z+\rho^2/z,\lambda)|_{z=1} = F(1+\rho^2,\lambda) = 0$. The estimator has a pole at $z = 1$ and is therefore exact for constant inputs. Next, we show that it converges linearly with rate ρ. Using the separated system, the solutions to $0 = F(z+\rho^2/z, L)$ are the union of the solutions to $0 = F(z+\rho^2/z,\lambda)$ for $\lambda \in \text{eig}(L)$. Since all solutions to $0 = F(z,\lambda)$ are in $[-2\rho, 2\rho]$, each solution has the form $z = 2\rho\cos\theta$ for some $\theta \in [-\pi, \pi]$. Then we have

$$0 = F(2\rho\cos\theta, \lambda) = F(z+\rho^2/z,\lambda)|_{z=\rho\exp(j\theta)}$$

so all of the estimator poles are on $\rho\mathbb{T}$. Therefore, the output converges with rate ρ. □

We will use this result in Chapter 4 to design feedback estimators which are exact and have fast convergence rates.

CHAPTER 3

Asymptotic Mean Ergodicity Property of Estimators

In this chapter we analyze the convergence properties of estimators for dynamic average consensus when the input signals are constant and the communication graph is time-varying, modeling noisy communication channels where packets are dropped randomly. The graph process is assumed to be i.i.d. and connected and balanced on average, but need not be connected or balanced at each time step. While the random graphs prevent asymptotic convergence of each agent's estimate to the average of the input signals, here we pose a slightly weaker question: Under what conditions do the time-averaged values of the estimator outputs converge to the actual average of the inputs? With this property, estimator outputs can be lowpass filtered to obtain the correct average.

When the communication graph is time-varying, the output of the estimator converges to a random process. We want to characterize when the time-average of the process is equal to its statistical average, which is a property known as mean ergodicity. By calculating the statistics of the steady-state process, we can apply an ergodic theorem to determine whether the steady-state process is ergodic in the mean. For estimators with this property, which we call *asymptotic mean ergodicity*, a local lowpass filter can be applied to the output of each agent to recover the global average of the input signals.

To study the steady-state behavior of estimators over random graphs, we first state an ergodic theorem which gives necessary and sufficient conditions on the steady-state covariance for a discrete-time random process to have the asymptotic mean ergodicity property.

For estimators with this property, the time-average of the output converges to the ensemble average of the steady-state process. We then derive an expression for the steady-state covariance of the output of a polynomial linear protocol. We then apply the ergodic theorem to the polynomial linear protocol to identify conditions under which the estimator is ergodic.

We first analyze a general one-hop polynomial linear protocol, and then apply these results to several specific estimators.

The subsequent sections are organized as follows. We define the covariance using the Kronecker product in Section 3.1 and establish several useful results of the covariance. In Section 3.2, we give sufficient conditions on the steady-state covariance for a random process to be asymptotically mean ergodic. We present our main theorem in Section 3.3 which develops the steady-state covariance for a general polynomial linear protocol using the separated system. We then apply these results to the P and PI estimators in Section 3.4 and summarize the results in Section 3.5.

3.1. Kronecker Product Covariance

Given random matrices $A \in \mathbb{R}^{m\times n}$ and $B \in \mathbb{R}^{p\times q}$, we would like a way of representing the covariances between $A_{i,j}$ and $B_{k,l}$ for all i, j, k, l. The covariance matrix between two random vectors $x \in \mathbb{R}^n$ and $y \in \mathbb{R}^m$ is generally defined as the $n \times m$ matrix

$$C(x, y) := \mathrm{E}\big[(x - \mathrm{E}[x])\,(y - \mathrm{E}[y])^T\big]\,. \tag{3.1}$$

This definition was generalized to tensors in [**58**]. Similarly, we generalize this definition to matrices using the Kronecker product in Definition 16.

Definition 16. *The covariance between $A \in \mathbb{R}^{m \times n}$ and $B \in \mathbb{R}^{p \times q}$ is defined as*

$$\mathrm{COV}[A, B] := \mathrm{E}[(A - \mathrm{E}[A]) \otimes (B - \mathrm{E}[B])] \tag{3.2}$$

which has dimensions $mp \times nq$, and where $\otimes$ denotes the Kronecker product. The variance is the covariance between a matrix and itself and is denoted

$$\mathrm{VAR}[A] := \mathrm{COV}[A, A]\,. \tag{3.3}$$

In the vector case, the covariances using the Kronecker product definition and the standard definition are related by $\mathrm{COV}[x, y] = \mathrm{vec}[C(y, x)]$ where $\mathrm{vec}[A]$ is the vectorization [**59**] of a matrix formed by stacking the columns of A. Equation (3.1) is valid only for random vectors while (3.2) is valid for both random vectors and matrices of any size, so we use the latter definition. Theorems 6 and 7 provide useful results for the Kronecker product covariance which are analogous to those of the scalar case.

Theorem 6. *Given matrices $A \in \mathbb{R}^{m \times n}$, $B \in \mathbb{R}^{p \times q}$ and vectors $x \in \mathbb{R}^n$, $y \in \mathbb{R}^q$ where A and B are uncorrelated with x and y, the covariance of the product is*

$$\begin{aligned}
\mathrm{COV}[Ax, By] &= \mathrm{COV}[A, B]\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) + (\mathrm{E}[A] \otimes \mathrm{E}[B])\,\mathrm{COV}[x, y] + \mathrm{COV}[A, B]\,\mathrm{COV}[x, y] \\
&= \mathrm{COV}[A, B]\,\mathrm{E}[x \otimes y] + (\mathrm{E}[A] \otimes \mathrm{E}[B])\,\mathrm{COV}[x, y] \\
&= \mathrm{COV}[A, B]\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) + \mathrm{E}[A \otimes B]\,\mathrm{COV}[x, y]\,. \qquad (3.4)
\end{aligned}$$

Proof.

$$
\begin{aligned}
&\mathrm{COV}[Ax, By] = \mathrm{E}[Ax \otimes By] - \mathrm{E}[Ax] \otimes \mathrm{E}[By] \\
&= \mathrm{E}[(A \otimes B)(x \otimes y)] - \mathrm{E}[A]\,\mathrm{E}[x] \otimes \mathrm{E}[B]\,\mathrm{E}[y] \\
&= \mathrm{E}[A \otimes B]\,\mathrm{E}[x \otimes y] - (\mathrm{E}[A] \otimes \mathrm{E}[B])\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) \\
&= (\mathrm{E}[A \otimes B] - \mathrm{E}[A] \otimes \mathrm{E}[B])\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) + (\mathrm{E}[A] \otimes \mathrm{E}[B])\,(\mathrm{E}[x \otimes y] - \mathrm{E}[x] \otimes \mathrm{E}[y]) \\
&\quad + (\mathrm{E}[A \otimes B] - \mathrm{E}[A] \otimes \mathrm{E}[B])\,(\mathrm{E}[x \otimes y] - \mathrm{E}[x] \otimes \mathrm{E}[y]) \\
&= \mathrm{COV}[A, B]\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) + (\mathrm{E}[A] \otimes \mathrm{E}[B])\,\mathrm{COV}[x, y] + \mathrm{COV}[A, B]\,\mathrm{COV}[x, y] \\
&= \mathrm{COV}[A, B]\,\mathrm{E}[x \otimes y] + (\mathrm{E}[A] \otimes \mathrm{E}[B])\,\mathrm{COV}[x, y] \\
&= \mathrm{COV}[A, B]\,(\mathrm{E}[x] \otimes \mathrm{E}[y]) + \mathrm{E}[A \otimes B]\,\mathrm{COV}[x, y]\,.
\end{aligned}
$$

□

For scalar random variables A, B, x, y, this result reduces to the standard result

$$\mathrm{COV}[Ax, By] = \mathrm{COV}[A, B]\,\mathrm{E}[x]\,\mathrm{E}[y] + \mathrm{E}[A]\,\mathrm{E}[B]\,\mathrm{COV}[x, y] + \mathrm{COV}[A, B]\,\mathrm{COV}[x, y]\,. \tag{3.5}$$

Theorem 7. *Given sets of vectors $x_i \in \mathbb{R}^m$ and $y_j \in \mathbb{R}^n$, the covariance of the sum is*

$$\mathrm{COV}\left[\sum_i x_i, \sum_j y_j\right] = \sum_i \sum_j \mathrm{COV}[x_i, y_j]\,. \tag{3.6}$$

Proof. This is a straightforward extension of the covariance of sums of scalars. □

We also use the following definition to simplify notation.

Definition 17. *Let k be a non-negative integer. The k^{th} Kronecker power is defined as*

$$X^{\otimes k} \equiv \underbrace{X \otimes \ldots \otimes X}_{k}\,. \tag{3.7}$$

3.2. Asymptotic Mean Ergodicity of Random Processes

Consider a discrete-time random process $\{X_k\}_{k=k_0}^{\infty}$ where $X_k \in \mathbb{R}^n$ for $k \geq k_0$. We extend the ergodic theorem for a wide-sense stationary random process in [**60**] to an asymptotically wide-sense stationary random process. This establishes conditions under which the time average of the process converges to the ensemble average as k approaches infinity.

Definition 18 (Asymptotically wide-sense stationary)**.** *The process X_k is asymptotically wide-sense stationary if and only if the mean and covariance of the steady-state process do not change with time; that is, the limits*

$$m_X := \lim_{n\to\infty} \mathrm{E}[X_n] \qquad \textit{and} \qquad C_X(k) := \lim_{n\to\infty} \mathrm{COV}[X_{k+n}, X_n] \tag{3.8}$$

exist and are finite where m_X is the mean and $C_X(k)$ is the covariance of the steady-state process.

Definition 19 (Time average)**.** *The time-average mean of a random process X_k starting at k_0 is given by*

$$\langle X_n \rangle_T = \frac{1}{T} \sum_{k=k_0}^{T+k_0-1} X_{k+n}. \tag{3.9}$$

We now state conditions under which the process is asymptotically ergodic in the mean.

Theorem 8 (Asymptotic mean ergodicity)**.** *Let $\{X_k\}_{k=k_0}^{\infty}$ be a single-sided asymptotically wide-sense stationary discrete-time random process with limiting mean m_X and limiting covariance $C_X(k)$. The process is asymptotically mean ergodic, that is,*

$$\lim_{T\to\infty} \lim_{n\to\infty} \langle X_n \rangle_T = m_X, \tag{3.10}$$

in the mean square sense if and only if the quantity

$$\text{AME}(X) := \lim_{T\to\infty} \frac{1}{T} \sum_{k=-(T-1)}^{T-1} \left(1 - \frac{|k|}{T}\right)\ C_X(k) \tag{3.11}$$

is zero.

Proof. Similar to [**60**], the limiting variance of $\langle X_k \rangle_T$ is

$$\begin{aligned}
\lim_{n\to\infty} \text{VAR}[\langle X_n\rangle_T] &= \lim_{n\to\infty} \text{E}\left[\left(\frac{1}{T}\sum_{k=k_0}^{T+k_0-1}\big(X_{k+n} - E[X_{k+n}]\big)\right)\left(\frac{1}{T}\sum_{l=k_0}^{T+k_0-1}\big(X_{l+n} - E[X_{l+n}]\big)\right)^T\right] \\
&= \frac{1}{T^2}\sum_{k=k_0}^{T+k_0-1}\sum_{l=k_0}^{T+k_0-1} \lim_{n\to\infty} \text{COV}[X_{k+n}, X_{l+n}] \\
&= \frac{1}{T^2}\sum_{k=0}^{T-1}\sum_{l=0}^{T-1} C_X(k-l) \\
&= \frac{1}{T^2}\sum_{k=-(T-1)}^{T-1} (T - |k|)\ C_X(k) \\
&= \frac{1}{T}\sum_{k=-(T-1)}^{T-1} \left(1 - \frac{|k|}{T}\right)\ C_X(k). \qquad (3.12)
\end{aligned}$$

Thus, if the limiting correlation goes to zero as

$$\lim_{T\to\infty} \frac{1}{T} \sum_{k=-(T-1)}^{T-1} \left(1 - \frac{|k|}{T}\right)\ C_X(k) = 0, \tag{3.13}$$

then the limiting time average converges to the limiting sample average in the mean square sense,

$$\lim_{T\to\infty}\lim_{n\to\infty} \text{E}\Big[\big(\langle X_n\rangle_T - \text{E}[\langle X_n\rangle_T]\big)\big(\langle X_n\rangle_T - \text{E}[\langle X_n\rangle_T]\big)^T\Big] = 0. \tag{3.14}$$

□

Corollary 1. *An asymptotically wide-sense stationary random process X_k with steady-state covariance given by*

$$C_X(k) = \lambda^{|k|} \tag{3.15}$$

is asymptotically mean ergodic if and only if $|\lambda| \leq 1$ and $\lambda \neq 1$.

Proof. For the given covariance function, (3.11) evaluates to

$$\text{AME}(X) = \begin{cases} 0, & |\lambda| \leq 1 \text{ and } \lambda \neq 1 \\ 1, & \lambda = 1 \\ \infty, & \text{otherwise.} \end{cases} \qquad \square$$

Definition 20 (Convergent)**.** *A square matrix A is convergent when its power sequence $\{A^k\}_{k=1}^{\infty}$ converges to a finite constant matrix as $k \to \infty$. From the Jordan decomposition, A is convergent if and only if all Jordan blocks associated with eigenvalues at $\lambda = 1$ are of size one, and all other eigenvalues have magnitude less than one.*

Corollary 2. *An asymptotically wide-sense stationary random process X_k with steady-state covariance given by*

$$C_X(k) = CA^{|k|}B \tag{3.16}$$

where $A \in \mathbb{R}^{n\times n}$ is convergent, $B \in \mathbb{R}^{n\times 1}$, $C \in \mathbb{R}^{1\times n}$, and any eigenvalue of A at one is either uncontrollable through B or unobservable through C, is asymptotically mean ergodic.

Proof. Perform the decomposition $A = P^{-1}JP$ where P is invertible and J is in Jordan form. Then $J = \text{diag}(J_1, \ldots, J_n)$ where each J_i, $i = 1, \ldots, n$ is a Jordan block of size m_i

with eigenvalue λ_i. Then we have

$$J_i^k = \begin{bmatrix} \lambda_i^k & k\frac{\lambda_i^{k-1}}{1!} & \dots & \frac{k!}{[k-(m_i-1)]!}\frac{\lambda_i^{k-(m_i-1)}}{(m_i-1)!} \\ 0 & \lambda_i^k & \dots & \frac{k!}{[k-(m_i-2)]!}\frac{\lambda_i^{k-(m_i-2)}}{(m_i-2)!} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & \lambda_i^k \end{bmatrix}. \tag{3.17}$$

where $\binom{n}{k} = n!/[(n-k)!k!]$. Each term in (3.11) corresponding to a Jordan block J_i with $|\lambda_i| < 1$ is zero by Corollary 1. All other Jordan blocks have $\lambda_i = 1$ and $m_i = 1$, so $J_i = 1$. If the eigenvalue is uncontrollable through B or unobservable through C, then the respective entry of B or C corresponding to J_i is zero. Thus (3.11) evaluates to zero, so the process is asymptotically mean ergodic by Theorem 8. □

3.3. Asymptotic Mean Ergodicity of Estimators

To analyze the asymptotic mean ergodicity property of estimators, we represent the estimator as a polynomial linear protocol from Definition 12. In this section, we present our main theorem which provides conditions under which the output of a polynomial linear protocol is asymptotically mean ergodic by examining the covariance of the output process. Ergodicity implies that time averages are equal to ensemble averages, so the low-pass filtered output of each agent converges to the ensemble average as time approaches infinity if the process is asymptotically mean ergodic. For one-hop estimators, Corollary 3 shows that the expected output is the output of the deterministic system using the expected Laplacian.

We now state our main theorem which gives conditions under which an estimator is asymptotically mean ergodic.

Theorem 9. *Let $\mathcal{G}$ be a time-varying graph with Laplacian matrix L_k at iteration k. Assume that $\mathrm{E}[L_k]$ is balanced and connected, and L_k are i.i.d. and independent of the initial state for all k. Consider an r-hop polynomial linear protocol $\Sigma(L_k)$. The output process due to a constant input is asymptotically mean ergodic if the following hold:*

(1) A_0 is convergent,

(2) any eigenvalues of A_0 with magnitude one are unobservable through C_0,

(3) $\rho\big(E[A(\hat{L}_k)]\big) < 1$, and

(4) $C_i = D_i = 0$ for $0 < i \leq r$.

Remark 2. *The assumptions on the Laplacian in Theorem 9 do not require the Laplacian to be balanced or connected at any individual time step.*

Remark 3. *Requirements (1) and (2) in Theorem 9 are also necessary for the estimator Σ to be robust to initial conditions* [**15**]. *Requirement (3) eliminates the possibility of $E[A(\hat{L}_k)]$ having simple eigenvalues at one, and requirement (4) is needed to evaluate the expression for the covariance of the output.*

Proof of Theorem 9. Similar to Section 2.6, we can separate the system into the consensus and disagreement directions. Let $Q \in \mathbb{R}^{n\times n}$ be an orthogonal matrix such that $Q = [v \; S]$ where $v = \mathbf{1}_n/\sqrt{n}$ and $S \in \mathbb{R}^{n\times(n-1)}$. Define the reduced Laplacian to be $\hat{L} := S^T L S$. Since S and v are orthogonal, we have $v^{\scriptscriptstyle T} S = 0$. To simplify notation, let $\tilde{v} = v \otimes I$ and $\tilde{S} = S \otimes I$ so that $\tilde{Q} = Q \otimes I = [\tilde{v} \; \tilde{S}]$. Performing the change of variable

$\tilde{x}_k = \tilde{Q}^T x_k$, the separated system $\tilde{\Sigma}(L)$ is

$$\tilde{A}(L) = \tilde{Q}^T A(L) \tilde{Q} = \begin{bmatrix} \tilde{v}^T A(L) \tilde{v} & \tilde{v}^T A(L) \tilde{S} \\ \tilde{S}^T A(L) \tilde{v} & \tilde{S}^T A(L) \tilde{S} \end{bmatrix} = \begin{bmatrix} A_0 & \tilde{v}^T A(L) \tilde{S} \\ 0 & A(\hat{L}) \end{bmatrix} \tag{3.18}$$

$$\tilde{B}(L) = \tilde{Q}^T B(L) = \begin{bmatrix} \tilde{v}^T B(L) \\ \tilde{S}^T B(L) \end{bmatrix} \tag{3.19}$$

$$\tilde{C}(L) = C(L) \tilde{Q} = \begin{bmatrix} C(L)\tilde{v} & C(L)\tilde{S} \end{bmatrix} = \begin{bmatrix} v \otimes C_0 & C(L)\tilde{S} \end{bmatrix} \tag{3.20}$$

$$\tilde{D}(L) = D(L) \tag{3.21}$$

which is equivalent to the original system. The system is then

$$\begin{bmatrix} z_{k+1} \\ w_{k+1} \end{bmatrix} = \begin{bmatrix} A_0 & \tilde{v}^T A(L_k) \tilde{S} \\ 0 & A(\hat{L}_k) \end{bmatrix} \begin{bmatrix} z_k \\ w_k \end{bmatrix} + \begin{bmatrix} \tilde{v}^T B(L_k) \\ \tilde{S}^T B(L_k) \end{bmatrix} u \tag{3.22}$$

$$y_k = \begin{bmatrix} v \otimes C_0 & C(L_k)\tilde{S} \end{bmatrix} \begin{bmatrix} z_k \\ w_k \end{bmatrix} + D(L_k) u \tag{3.23}$$

for $k \geq 0$ with initial conditions z_0 and w_0. We can also write the system using

$$\begin{aligned} A(L) &= \tilde{L}\tilde{A} & \qquad B(L) &= \tilde{L}\tilde{B} \\ C(L) &= \bar{L}\tilde{C} & \qquad D(L) &= \bar{L}\tilde{D} \end{aligned} \tag{3.24}$$

where

$$\tilde{L} = \begin{bmatrix} I_n \otimes I_p & L \otimes I_p & \dots & L^r \otimes I_p \end{bmatrix}, \tag{3.25}$$

$$\bar{L} = \begin{bmatrix} I_n & L & \dots & L^r \end{bmatrix} \tag{3.26}$$

and

$$\tilde{A} = \begin{bmatrix} I_n \otimes A_0 \\ \vdots \\ I_n \otimes A_r \end{bmatrix} \qquad \tilde{B} = \begin{bmatrix} I_n \otimes B_0 \\ \vdots \\ I_n \otimes B_r \end{bmatrix}$$

$$\tilde{C} = \begin{bmatrix} I_n \otimes C_0 \\ \vdots \\ I_n \otimes C_r \end{bmatrix} \qquad \tilde{D} = \begin{bmatrix} I_n \otimes D_0 \\ \vdots \\ I_n \otimes D_r \end{bmatrix}. \tag{3.27}$$

Define the steady-state covariance between two random vectors x_k and y_k as

$$\sigma^2_{x,y} \equiv \lim_{k \to \infty} \mathrm{COV}[x_k, y_k]\,. \tag{3.28}$$

We want to determine the steady-state covariance of the output in (3.23),

$$C_y(k) \equiv \lim_{n \to \infty} \mathrm{COV}[y_{n+k}, y_n]\,, \tag{3.29}$$

to determine if the system is asymptotically mean ergodic using Theorem 8. Note that z_k and w_k are uncorrelated with L_j for $j \geq k$. Using the Kronecker product relations from

Section 3.1, the following covariances are zero,

$$\mathrm{COV}\Big[A_0 z_k, \tilde{v}^T A(L_k)\tilde{S}w_k\Big] = (A_0 \otimes \tilde{v}^T \mathrm{E}\Big[\tilde{L}_k\Big]\,\tilde{A}\tilde{S})\mathrm{COV}[z_k, w_k] = 0 \tag{3.30}$$

and

$$\mathrm{COV}\big[A_0 z_k, \tilde{v}^T B(L_k)u\big] = (A_0 \otimes \tilde{v}^T)\mathrm{COV}\Big[z_k, \tilde{L}_k\Big]\,(I \otimes \tilde{B}u) = 0 \tag{3.31}$$

since $\mathrm{E}[L_k]$ is balanced and z_k is uncorrelated with L_k, respectively. Define the variance

$$\begin{aligned}\hat{\sigma}^2(k) &\equiv \mathrm{VAR}\Big[A(L_k)\tilde{S}w_k + B(L_k)u\Big] \\ &= \mathrm{VAR}\Big[\tilde{L}_k(\tilde{A}\tilde{S}w_k + \tilde{B}u)\Big] \\ &= \mathrm{E}\Big[\tilde{L}_k^{\otimes 2}\Big]\,(\tilde{A}\tilde{S})^{\otimes 2}\mathrm{VAR}[w_k] + \mathrm{VAR}\Big[\tilde{L}_k\Big]\,(\tilde{A}\tilde{S}\ \mathrm{E}[w_k] + \tilde{B}u)^{\otimes 2}.\end{aligned} \tag{3.32}$$

Taking the limit as $k \to \infty$ of (3.32),

$$\hat{\sigma}^2 \equiv \lim_{k\to\infty} \hat{\sigma}^2(k) = \mathrm{E}\Big[\tilde{L}_k^{\otimes 2}\Big]\,(\tilde{A}\tilde{S})^{\otimes 2}\sigma_w^2 + \mathrm{VAR}\Big[\tilde{L}_k\Big]\,(\tilde{A}\tilde{S}\bar{w} + \tilde{B}u)^{\otimes 2} \tag{3.33}$$

where $\bar{w} = \lim_{k\to\infty} w_k$. The variances and covariances of the state can be found recursively using the Kronecker product relations from Section 3.1. Using (3.30) and (3.31), the variance of z_{k+1} is

$$\mathrm{VAR}[z_{k+1}] = A_0^{\otimes 2}\mathrm{VAR}[z_k] + (\tilde{v}^T)^{\otimes 2}\hat{\sigma}_x^2(k). \tag{3.34}$$

Similarly, the other variances and covariances of the state are

$$\begin{aligned} \mathrm{VAR}[w_{k+1}] &= (\tilde{S}^T)^{\otimes 2}\hat{\sigma}_x^2(k) \\ &= \mathrm{E}\Big[A(\hat{L}_k)^{\otimes 2}\Big]\,\mathrm{VAR}[w_k] + (\tilde{S}^T)^{\otimes 2}\mathrm{VAR}\Big[\tilde{L}_k\Big]\,(\tilde{A}\tilde{S}\ \mathrm{E}[w_k] + \tilde{B}u)^{\otimes 2}, \qquad (3.35) \end{aligned}$$

$$\mathrm{COV}[z_{k+1}, w_{k+1}] = \Big(A_0 \otimes \mathrm{E}\Big[A(\hat{L}_k)\Big]\Big)\,\mathrm{COV}[z_k, w_k] + (\tilde{v}^T \otimes \tilde{S}^T)\hat{\sigma}_x^2(k), \qquad (3.36)$$

$$\mathrm{COV}[w_{k+1}, z_{k+1}] = \Big(\mathrm{E}\Big[A(\hat{L}_k)\Big] \otimes A_0\Big)\,\mathrm{COV}[w_k, z_k] + (\tilde{S}^T \otimes \tilde{v}^T)\hat{\sigma}_x^2(k). \qquad (3.37)$$

The steady-state variance of z_k may be infinite due to the possible eigenvalue of A_0 at one. The other systems all have eigenvalues with magnitude less than one, so the steady-state variances and covariances are given by

$$\sigma_w^2 = \Big[I - \mathrm{E}\Big[A(\hat{L}_k)^{\otimes 2}\Big]\Big]^{-1}(\tilde{S}^T)^{\otimes 2}\mathrm{VAR}\Big[\tilde{L}_k\Big]\,(\tilde{A}\tilde{S}\bar{w} + \tilde{B}\bar{u})^{\otimes 2}, \qquad (3.38)$$

$$\sigma_{z,w}^2 = \Big[I - A_0 \otimes \mathrm{E}\Big[A(\hat{L}_k)\Big]\Big]^{-1}(\tilde{v}^T \otimes \tilde{S}^T)\hat{\sigma}_x^2, \qquad (3.39)$$

$$\sigma_{w,z}^2 = \Big[I - \mathrm{E}\Big[A(\hat{L}_k)\Big] \otimes A_0\Big]^{-1}(\tilde{S}^T \otimes \tilde{v}^T)\hat{\sigma}_x^2. \qquad (3.40)$$

The covariance between the state at iteration $k+i$ and k is

$$\mathrm{COV}[z_{k+i}, z_k] = \big(A_0^i \otimes I\big)\,\mathrm{VAR}[z_k]\,, \qquad (3.41)$$

$$\mathrm{COV}[w_{k+i}, w_k] = \Big(\mathrm{E}\Big[A(\hat{L}_k)\Big]^i \otimes I\Big)\,\mathrm{VAR}[w_k]\,, \qquad (3.42)$$

$$\mathrm{COV}[z_{k+i}, w_k] = \big(A_0^i \otimes I\big)\,\mathrm{COV}[z_k, w_k]\,, \qquad (3.43)$$

$$\mathrm{COV}[w_{k+i}, z_k] = \Big(\mathrm{E}\Big[A(\hat{L}_k)\Big]^i \otimes I\Big)\,\mathrm{COV}[w_k, z_k]\,. \qquad (3.44)$$

For the steady-state process, the covariances are

$$\lim_{k\to\infty} \mathrm{COV}[z_{k+i}, z_k] = \left(A_0^i \otimes I\right)\sigma_z^2, \tag{3.45}$$

$$\lim_{k\to\infty} \mathrm{COV}[w_{k+i}, w_k] = \left(\mathrm{E}\Big[A(\hat{L}_k)\Big]^i \otimes I\right)\sigma_w^2, \tag{3.46}$$

$$\lim_{k\to\infty} \mathrm{COV}[z_{k+i}, w_k] = \left(A_0^i \otimes I\right)\sigma_{z,w}^2, \tag{3.47}$$

$$\lim_{k\to\infty} \mathrm{COV}[w_{k+i}, z_k] = \left(\mathrm{E}\Big[A(\hat{L}_k)\Big]^i \otimes I\right)\sigma_{w,z}^2. \tag{3.48}$$

Using the assumption that $C_i = D_i = 0$ for $0 < i \leq l$, the steady-state covariance of the output at time step $k + i$ and k is

$$\begin{aligned} C_y(i) &= \lim_{k\to\infty} \mathrm{COV}[y_{k+i}, y_k] && (3.49)\\ &= \lim_{k\to\infty} \left[(v_e \otimes C_0) \otimes (v_e \otimes C_0)\right] \mathrm{COV}[z_{k+i}, z_k] \\ &\quad + \left[(v_e \otimes C_0) \otimes (S \otimes C_0)\right] \mathrm{COV}[z_{k+i}, w_k] \\ &\quad + \left[(S \otimes C_0) \otimes (v_e \otimes C_0)\right] \mathrm{COV}[w_{k+i}, z_k] \\ &\quad + \left[(S \otimes C_0) \otimes (S \otimes C_0)\right] \mathrm{COV}[w_{k+i}, w_k]. && (3.50) \end{aligned}$$

Using the expressions in (3.45) to (3.48),

$$\begin{aligned} C_y(i) = {} & \left[(v_e \otimes C_0 A_0^i) \otimes (v_e \otimes C_0)\right] \sigma_z^2 \\ & + \left[(v_e \otimes C_0 A_0^i) \otimes (S \otimes C_0)\right] \sigma_{z,w}^2 \\ & + \left[(S \otimes C_0) \otimes (v_e \otimes C_0)\right] \left(\mathrm{E}\Big[A(\hat{L}_k)\Big] \otimes I\right)^i \sigma_{w,z}^2 \\ & + \left[(S \otimes C_0) \otimes (S \otimes C_0)\right] \left(\mathrm{E}\Big[A(\hat{L}_k)\Big] \otimes I\right)^i \sigma_w^2. \end{aligned} \tag{3.51}$$

Both A_0 and $\mathrm{E}[\hat{L}_k]$ are convergent. Any eigenvalues at one of A_0 are unobservable through C_0, and $\mathrm{E}[\hat{L}_k]$ has no eigenvalues at one. Therefore we can apply Corollary 2. The steady-state variance σ_z^2 may be infinite, however, which would cause the system to not be ergodic. σ_z^2 may only have infinite values in positions corresponding to an eigenvalue at one of A_0, but this does not affect the output since the eigenvalue at one is unobservable through C_0. Therefore the output y_k is ergodic by Corollary 2. □

Remark 4. *If C_i and D_i are nonzero for $0 < i \leq r$ in Theorem 9, then the steady-state covariance of the output is*

$$\begin{aligned} C_y(i) = \lim_{k\to\infty} & \mathrm{COV}[(v_e \otimes C_0)z_{k+i}, (v_e \otimes C_0)z_k] + \mathrm{COV}[(v_e \otimes C_0)z_{k+i}, D(L_k)u] \\ & + \mathrm{COV}[D(L_{k+i})u, (v_e \otimes C_0)z_k] + \mathrm{COV}\left[(v_e \otimes C_0)z_{k+i}, \tilde{S}C(\hat{L}_k)w_k\right] \\ & + \mathrm{COV}\left[\tilde{S}C(\hat{L}_{k+i})w_{k+i}, (v_e \otimes C_0)z_k\right] + \mathrm{COV}\left[\tilde{S}C(\hat{L}_{k+i})w_{k+i}, \tilde{S}C(\hat{L}_k)w_k\right] \\ & + \mathrm{COV}\left[\tilde{S}C(\hat{L}_{k+i})w_{k+i}, D(L_k)u\right] + \mathrm{COV}\left[D(L_{k+i})u, \tilde{S}C(\hat{L}_k)w_k\right] \\ & + \mathrm{COV}[D(L_{k+i})u, D(L_k)u]\,. \end{aligned} \tag{3.52}$$

For one-hop estimators, the system is linear in L_k which results in the following.

Corollary 3. *Under the same assumptions as Theorem 9, the time-averaged output of each agent converges to the same output as $\Sigma(\mathrm{E}[L_k])$ if the protocol is one-hop.*

Proof. Since the process is linear in L_k,

$$\begin{aligned} &\mathrm{E}[A(L_k)] = A(\mathrm{E}[L_k]) && \mathrm{E}[B(L_k)] = B(\mathrm{E}[L_k]) \\ &\mathrm{E}[C(L_k)] = C(\mathrm{E}[L_k]) && \mathrm{E}[D(L_k)] = D(\mathrm{E}[L_k]). \end{aligned} \tag{3.53}$$

The expected system is then

$$\mathrm{E}[x_{k+1}] = A(\mathrm{E}[L_k])\mathrm{E}[x_k] + B(\mathrm{E}[L_k])u_k \tag{3.54}$$

$$\mathrm{E}[y_k] = C(\mathrm{E}[L_k])\mathrm{E}[x_k] + D(\mathrm{E}[L_k])u_k. \tag{3.55}$$

since x_k is independent of L_k. The expected output is the output of $\Sigma(\mathrm{E}[L_k])$. Since the output process is ergodic by Theorem 9, the time-averaged output of each agent converges to the expected output. □

3.4. Examples

Our results hold for a general polynomial linear protocol. We now apply the results to the following two estimators.

Example 1. *The P estimator* [**19**] *is a one-hop polynomial linear protocol with parameters γ and k_p where*

$$\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \left[\begin{array}{c|c} 1-\gamma & 0 \\ \hline 1 & 1 \end{array}\right] \otimes I_n + \left[\begin{array}{c|c} -k_p & -k_p \\ \hline 0 & 0 \end{array}\right] \otimes L. \tag{3.56}$$

Example 2. *The PI estimator* [**19**] *is a one-hop polynomial linear protocol with parameters γ, k_p, and k_I where*

$$\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \left[\begin{array}{cc|c} 1-\gamma & 0 & \gamma \\ 1 & 0 & 0 \\ \hline 1 & 0 & 0 \end{array}\right] \otimes I_n + \left[\begin{array}{cc|c} -k_p & k_I & 0 \\ -k_I & 0 & 0 \\ \hline 0 & 0 & 0 \end{array}\right] \otimes L. \tag{3.57}$$

Both the P and PI estimators are one-hop polynomial linear protocols, so asymptotic mean ergodicity implies that the time-averaged output of each agent converges to the output of the estimator using the expected Laplacian by Corollary 3. The expected steady-state error and robustness to initial conditions is known for the deterministic system [**15**]. The ergodicity property of each estimator is discussed below, and the results are shown in Table 3.1.

Table 3.1. Summary of properties for the P and PI estimators with $\mathrm{E}[L_k]$ balanced and connected, and L_k i.i.d. for all k.

Estimator	**Ergodic**	**Robust to ICs**	$\lim_{k\to\infty} \mathrm{E}[e_k] = 0$
P, $\gamma = 0$	✗	✗	✓[1]
P, $\gamma \neq 0$	✓	✓	✗
PI	✓	✓	✓

[1] If the expectation of the initial state sums to zero.

3.4.1. P Estimator

Consider the P estimator in Example 1. The estimator has different properties depending on the forgetting factor γ since A_0 has an eigenvalue at one if $\gamma = 0$. We analyze the two cases separately.

3.4.1.1. Case 1: $\gamma \neq 0$. In this case, the eigenvalue of A_0 is $1 - \gamma$ which must have magnitude less than one, so we require $0 < \gamma < 2$. In order for the eigenvalues of $\mathrm{E}[A(\hat{L}_k)]$ to have magnitude less than one, k_p must be chosen such that $|(1-\gamma) - k_p\lambda_i| < 1$ for each eigenvalue λ_i of $\mathrm{E}[\hat{L}_k]$. If both constraints are satisfied, then the four conditions in Theorem 9 are satisfied so the protocol is asymptotically mean ergodic. The expected output, however, does not converge to the correct average with zero steady-state error. Therefore the time average converges to the ensemble average, but neither is the average of the input signals.

3.4.1.2. Case 2: $\gamma = 0$. In this case, the pair (A_0, C_0) has an observable eigenvalue at one, so the estimator is not ergodic. The system can be made ergodic, however, with extra restrictions. If the Laplacian is balanced at each time step, then $\sigma^2_{z,w} = 0$. In addition, $\sigma^2_z = 0$ if $\mathrm{VAR}[z_0] = 0$. Using (3.51), the output is ergodic if L_k is balanced for all k and $\mathrm{VAR}[z_0] = 0$. In this case the estimator does converge to the average of the inputs, so ergodicity gives that the time-averaged output of each agent converges to the average of the inputs, but this requires the extra restrictions on the Laplacian and initial state.

3.4.2. PI Estimator

Consider the PI estimator in Example 2. The eigenvalue at one of A_0 is unobservable through C_0, and the other eigenvalue of A_0 is $1-\gamma$ so we require $0 < \gamma < 2$. The constants k_p and k_I must be chosen such that the eigenvalues of $A(\mathrm{E}[\hat{L}_k])$ have magnitude less than one. If these conditions are satisfied, then the output y_k is asymptotically mean ergodic by Theorem 9, so the time-averaged output of each agent converges to the ensemble average. Since the expected steady-state error is zero independent of initial conditions, the time-averaged output of each agent converges to the average of the inputs for any initial conditions.

3.5. Summary

We studied the convergence properties of the P and PI estimators when packets are dropped at random. Dropped packets were modeled by i.i.d. random Laplacians assumed to be balanced and connected on average, although the Laplacian need not be balanced or connected at any time step. This model is limited to situations in which the packet drop probabilities are symmetric between two agents, so that the expected Laplacian is balanced.

To study the convergence properties of average consensus estimators over random graphs, the covariance was defined using the Kronecker product in order to obtain an expression for the steady-state covariance of the system output. An ergodic theorem then gave conditions under which the output of a polynomial linear protocol is asymptotically mean ergodic. Results were applied to the P and PI estimators, and it was shown that the P estimator is ergodic either if $\gamma \neq 0$, or the Laplacian is balanced at each time step and the initial state is deterministic. The PI estimator, however, is always ergodic (provided that γ, k_p, k_I satisfy the conditions in Section 3.4.2), has zero expected steady-state error, and is robust to initial conditions. For the PI estimator, the time-averaged output of each agent converges with zero steady-state error in the presence of dropped packets independent of initial conditions.

CHAPTER 4

Estimators for Signals with Discrete Frequency Spectrum

In this chapter we design estimators to solve the dynamic average consensus problem when the frequency spectrum of the input signals is composed of discrete frequencies. For example, the input signal for agent i could be any of the following:

$$u_i^k = \begin{cases} u_i, & \text{constant} \\ u_{i,0} + u_{i,1}\,k + u_{i,2}\,k^2 + \ldots + u_{i,m}\,k^m, & \text{polynomial with degree } m \\ u_i \cos(\omega k - \theta_i), & \text{sinusoidal with frequency } \omega \end{cases} \tag{4.1}$$

with corresponding z-transforms

$$u_i(z) = \begin{cases} \dfrac{z}{z-1} u_i, & \text{constant} \\ \dfrac{u_{i,0}\,z + u_{i,1}\,z^2 + u_{i,2}\,z^3 + \ldots + u_{i,m}\,z^{m+1}}{(z-1)^{m+1}}, & \text{polynomial with degree } m \\ \dfrac{z(z\cos(\theta_i) - \cos(\omega + \theta_i))}{z^2 - 2z\cos(\omega) + 1} u, & \text{sinusoidal with frequency } \omega. \end{cases} \tag{4.2}$$

We refer to the denominator of the z-transform as the *model* of the signal. Throughout this chapter we assume that each of the input signals has the same model, and that the model is known. If the signals are known to be sinusoidal but with unknown frequency, then the frequency may first be estimated and the estimate used to design the estimator [**22**]. For simplicity we design the estimators to track constant input signals with zero steady-state error. However, all of the designs can be extended to track other types of signals as long as the model is known. For models with high-degree (e.g., signals with multiple frequencies or high-degree polynomials), we factor the model as $d(z) = d_1(z)\, d_2(z) \dots d_m(z)$. Multiple estimators are then cascaded together in series where the i^{th} stage is designed using $d_i(z)$ as the model.

For input signals with discrete frequency spectrum, a model of the inputs can be placed in the feedback loop to achieve zero steady-state error. Therefore, all of the estimators in this chapter are feedback designs.

Throughout this chapter we assume that the communication graph is constant and undirected. Furthermore, we assume knowledge of lower and upper bounds on the nonzero eigenvalues of the graph Laplacian, denoted $\lambda_{\min}$ and $\lambda_{\max}$, respectively. The lower bound is soft in the sense that if it is violated, then our estimators will still converge but without any guarantees on their rates of convergence. We accept such a soft bound as it may be difficult to obtain a non-conservative hard lower bound on the nonzero eigenvalues of L. As we will see below, the convergence rates of our estimators will depend on the ratio $\lambda_r = \lambda_{\min}/\lambda_{\max}$. See Section 2.2.1 for how to choose the edge weights to either maximize λ_r (if the graph topology is known) or obtain the bounds $\lambda_{\min}$ and $\lambda_{\max}$ (if the graph topology is unknown).

Therefore, we solve the following problem in this chapter.

Problem 3. *Suppose $\mathcal{G}$ is a constant, connected, and undirected graph whose Laplacian matrix has nonzero eigenvalues in the interval $[\lambda_{\min}, \lambda_{\max}]$ with $0 < \lambda_{\min} \leq \lambda_{\max}$. Also, suppose the input signals have a known model, i.e., $d(z)$ is known where $u(z) = n(z)/d(z)$ with $u(z)$ and $d(z)$ polynomials in z. Design an estimator which*

(1) uses one-hop local broadcast communication,

(2) is scalable,

(3) is internally stable,

(4) is time-invariant,

(5) and has error that converges to zero linearly with rate $\rho < 1$ from any initial state.

Furthermore, the estimator should have the minimum attainable value of ρ.

4.1. Static Estimator

Perhaps the simplest estimator for dynamic average consensus is the *static estimator* which has been studied at least as far back as [**11**]. The term "static" is used to describe the estimator because it is only capable of tracking input signals which are constant, i.e., $u_k = u$ for all $k \geq 0$. To implement the static estimator, agent i uses the equations

$$y^i_{k+1} = y^i_k - k_p \sum_{j \in \mathcal{N}_i} a_{ij}(y^i_k - y^j_k), \quad y^i_0 = u^i \tag{4.3}$$

where y_k is the estimate of the average at time k and k_p is a parameter. By stacking the agent variables into vectors, we can write this compactly as

$$y_{k+1} = (I - k_p L) y_k, \quad y_0 = u. \tag{4.4}$$

The state-space representation is

$$\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \left[\begin{array}{c|c} 1 & 0 \\ \hline 1 & 0 \end{array}\right] \otimes I + \left[\begin{array}{c|c} -k_p & 0 \\ \hline 0 & 0 \end{array}\right] \otimes L \tag{4.5}$$

and the block diagram is given in Fig. 4.1. Note that the input u is not a signal but a constant initial state, so this estimator is only capable of tracking the average of constant signals.

A standard convergence result for this estimator is given below.

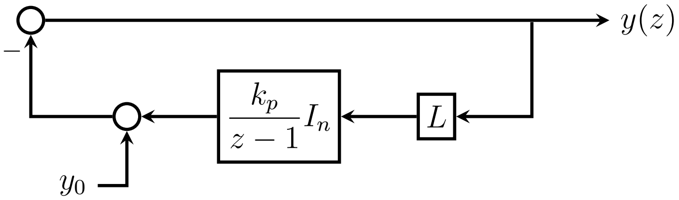


Figure 4.1. Block diagram of the static estimator. The input signals are assigned to the initial conditions, i.e., $y_0 = u$.

Theorem 10 (Static estimator [**12**, Thm. 1])**.** *Suppose that $\mathcal{G}$ is constant, connected, balanced, and has Laplacian matrix which satisfies $\rho := \rho(I_n - k_pL - \Pi_n) < 1$, and that the input signals are constant. Then the error of the static estimator converges to zero linearly with rate ρ.*

4.1.1. Optimizing the convergence rate

From Theorem 10, we want to design the estimator such that $\rho := \rho(I_n - k_pL - \Pi_n) < 1$.

First, consider the case when the graph topology is known. Then we can choose the parameter k_p and the edge weights a_{ij} in a centralized manner to minimize ρ. If the graph is undirected, then the Laplacian is symmetric so $\rho = \|I_n - k_pL - \Pi_n\|$. The problem is convex in this case so we can set $k_p = 1$ and solve the convex optimization problem in Problem 2 to calculate the optimal weights. When the graph is directed, the problem is no longer convex [**56**]. We can still solve for the weights from Problem 2 since $\rho(I - k_pL - \Pi) \leq \|I - k_pL - \Pi\|$, although they are not optimal in this case.

We now consider the case when the graph topology is unknown. A simple choice for k_p which statisfies the condition $\rho(I - k_pL - \Pi) < 1$ and does not require a lower bound on the nonzero eigenvalues of L is $0 < k_p < 2/\|L\|$, and a standard choice is $k_p = 1/\|L\|$. If a lower bound is available and the graph is undirected, a better choice is

$$k_p = \frac{2}{\lambda_{\min} + \lambda_{\max}}. \tag{4.6}$$

This choice optimizes the worst-case convergence rate in the sense that it minimizes the maximum modulus of the non-unity eigenvalues of $I - k_pL$ over all Laplacian matrices L having nonzero eigenvalues in the interval $[\lambda_{\min}, \lambda_{\max}]$. The resulting convergence rate is

$$\rho = \frac{1 - \lambda_r}{1 + \lambda_r} \tag{4.7}$$

where $\lambda_r = \lambda_{\min}/\lambda_{\max}$. To see why this choice of k_p is used, consider the characteristic equation of the static estimator,

$$0 = zI - (I - k_pL). \tag{4.8}$$

Separating the system according to the eigenvalues of L gives

$$0 = 1 + \lambda\frac{k_p}{z - 1}. \tag{4.9}$$

To observe how the roots move as a function of the parameter λ, we use the root locus. For (4.9), the root locus is shown in Fig. 4.2. This shows how the poles of the estimator move as a function of the Laplacian eigenvalues λ. Note that ρ is the maximum magnitude of the closed-loop poles corresponding to nonzero eigenvalues of L. Since all nonzero eigenvalues are in the interval $[\lambda_{\min}, \lambda_{\max}]$, we want to choose k_p to minimize ρ such that the root locus is inside $\rho\mathbb{T}$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. This is achieved when the pole is at $z = \rho$ when $\lambda = \lambda_{\min}$

and at $z = -\rho$ when $\lambda = \lambda_{\max}$, i.e.,

$$\rho = 1 - k_p\lambda_{\min} \tag{4.10}$$

$$-\rho = 1 - k_p\lambda_{\max}. \tag{4.11}$$

Solving these conditions for k_p and ρ gives the values in (4.6) and (4.7), respectively.

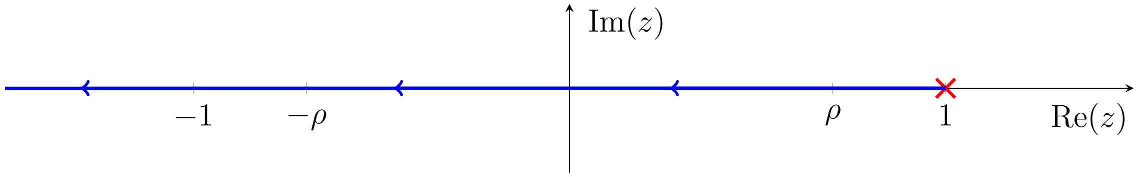


Figure 4.2. Root locus of the static estimator.

4.2. Dynamic Estimators with One Transmission Variable (P Estimator)

The static estimator in the previous section cannot track time-varying signals since the signals only enter the system as initial conditions. In contrast, *dynamic estimators*, introduced by [**16**], continuously inject the signals as inputs into the dynamical system. Dynamic estimators for dynamic average consensus in discrete time have been considered in [**15, 21, 23, 26, 27, 30, 31, 61, 62**].

In this section, we consider dynamic estimators which have each agent broadcast a single scalar variable to neighboring agents at each time step. Estimators with one variable transmitted at each iteration are characterized by their block diagrams having only one Laplacian block. For example, the static estimator in the previous section has one transmission variable per iteration, and the estimator can be made dynamic by applying the signals as inputs as shown in Fig. 4.3. This is sometimes referred to as the *proportional (P) estimator* [**19**].

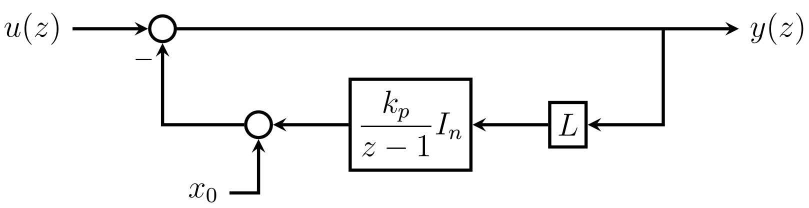
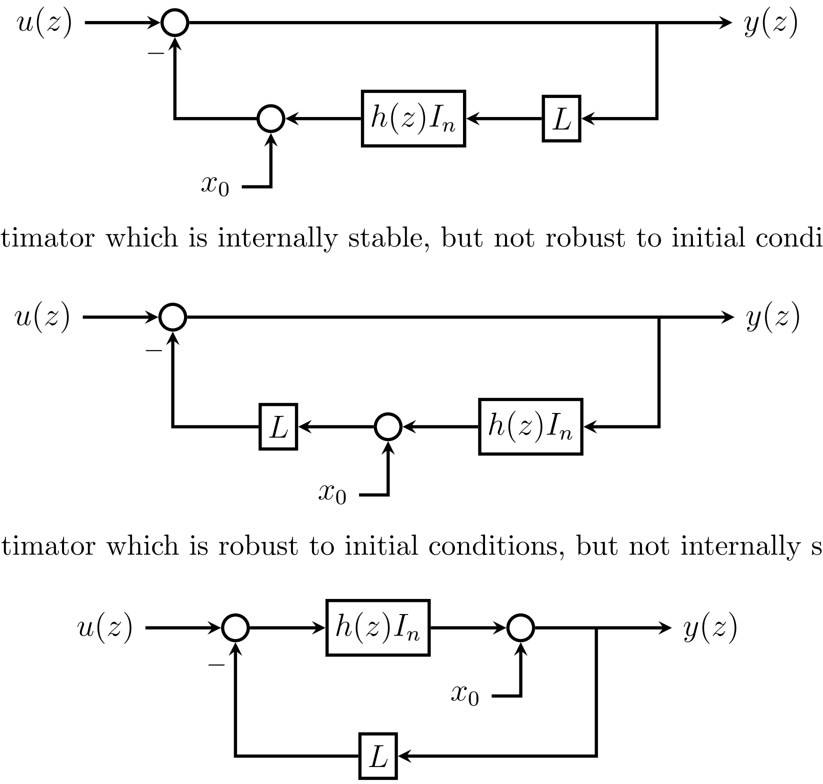


Figure 4.3. Block diagram of the dynamic version of the static estimator where the signals are applied as inputs to the system.

One can show that if $k_p\|L\| < 2$, if $\mathcal{G}$ is connected and balanced, and if the inputs are *constant* (i.e., $u_k = u$ for all $k \geq 0$), then as $k \to \infty$, all entries of the output vector y_k converge to $\text{avg}(x_0) + \text{avg}(u)$, where $\text{avg}(x_0)$ denotes the average of the entries in the initial state vector x_0. In other words, the steady-state error vector under constant inputs is just $\text{avg}(x_0)\mathbf{1}_n$. Thus if $\text{avg}(x_0) = 0$, the steady-state error will be zero for constant inputs and small for slowly-varying inputs. Furthermore, this estimator can be cascaded to achieve zero steady-state error for inputs that are polynomial functions of the discrete-time variable k: simply feed the output of the first stage into the input of the second, and so on as needed. As explained in [**23**], a cascade of m estimator stages will achieve zero steady-state error for polynomial inputs of degree up to $m - 1$. However, this estimator in Fig. 4.3 lacks initialization robustness: there is no open set of initial states that all lead to the same steady-state estimates. In other words, this estimator is not robust to initial conditions.

Note that here we inject u_k at the output of the integrator block and take the output y_k at the input to the Laplacian block. Other dynamic feedback estimators with different properties can be obtained by changing the orders of the blocks as well as where the inputs and outputs are taken. For example, the estimator in Fig. 4.4b switches the order of the Laplacian and integrator blocks, and the estimator in Fig. 4.4c applies the input before the integrator. Each of these estimators has different properties as described in Fig. 4.4. Also, instead of using a pure integrator $1/(z-1)$, each agent can apply any strictly proper transfer function $h(z)$. In order for the estimator to achieve zero steady-state error, $h(z)$ must be in the feedback loop and contain a pole at $z = 1$.



(a) Estimator which is internally stable, but not robust to initial conditions.

(b) Estimator which is robust to initial conditions, but not internally stable.

(c) Estimator which is neither robust to initial conditions nor internally stable.

Figure 4.4. Block diagrams of proportional estimators.

The filter $h(z)$ must be strictly proper so that Fig. 4.4 does not contain algebraic loops, so we can write

$$h(z) = \left[\begin{array}{c|c} a & b \\ \hline c & 0 \end{array}\right]. \tag{4.12}$$

The state-space representation of each of the estimators is then

$$\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \begin{cases} \left[\begin{array}{c|c} a & 0 \\ \hline -c & 1 \end{array}\right] \otimes I + \left[\begin{array}{c|c} -bc & b \\ \hline 0 & 0 \end{array}\right] \otimes L, & \text{Fig. 4.4a} \\ \left[\begin{array}{c|c} a & b \\ \hline 0 & 1 \end{array}\right] \otimes I + \left[\begin{array}{c|c} -bc & 0 \\ \hline -c & 0 \end{array}\right] \otimes L, & \text{Fig. 4.4b} \\ \left[\begin{array}{c|c} a & b \\ \hline c & 0 \end{array}\right] \otimes I + \left[\begin{array}{c|c} -bc & 0 \\ \hline 0 & 0 \end{array}\right] \otimes L, & \text{Fig. 4.4c} \end{cases} \tag{4.13}$$

and the estimators are implemented on agent i using the equations

$$\begin{aligned} x^i_{k+1} &= a\,x^i_k + k_p\,b \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(y^i_k - y^j_k) \\ y^i_k &= u^i_k - c\,x^i_k \end{aligned} \qquad \text{Fig. 4.4a} \tag{4.14}$$

$$\begin{aligned} x^i_{k+1} &= a\,x^i_k + k_p\,b \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(y^i_k - y^j_k) \\ y^i_k &= u^i_k - k_p\,c \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(x^i_k - x^j_k) \end{aligned} \qquad \text{Fig. 4.4b} \tag{4.15}$$

$$\begin{aligned} x^i_{k+1} &= a\,x^i_k - k_p\,b\,c \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(x^i_k - x^j_k) + b\,u^i_k \\ y^i_k &= c\,x^i_k \end{aligned} \qquad \text{Fig. 4.4c} \tag{4.16}$$

where $x^i_0 \in \mathbb{R}^p$ with $a \in \mathbb{R}^{p \times p}$, $b \in \mathbb{R}^p$, and $c^T \in \mathbb{R}^p$.

Each of the estimators in Fig. 4.4 have different properties which depend on the filter $h(z)$. In order for the estimator to have zero steady-state error for constant inputs, there must be an integrator in the feedback loop. This is satisfied for the estimators in Figs. 4.4a and 4.4b if $h(z)$ has a pole at $z = 1$. This can be generalized to other classes of inputs with a given model, i.e., the inputs are $u(z) = n(z)/d(z)$ where n and d are polynomials and $d(z)$ is known. In that case, the estimators achieve zero steady-state error if $h(z)$ contains a model of the input, i.e., $d(z)$ divides $d_h(z)$ where $h(z) = n_h(z)/d_h(z)$.

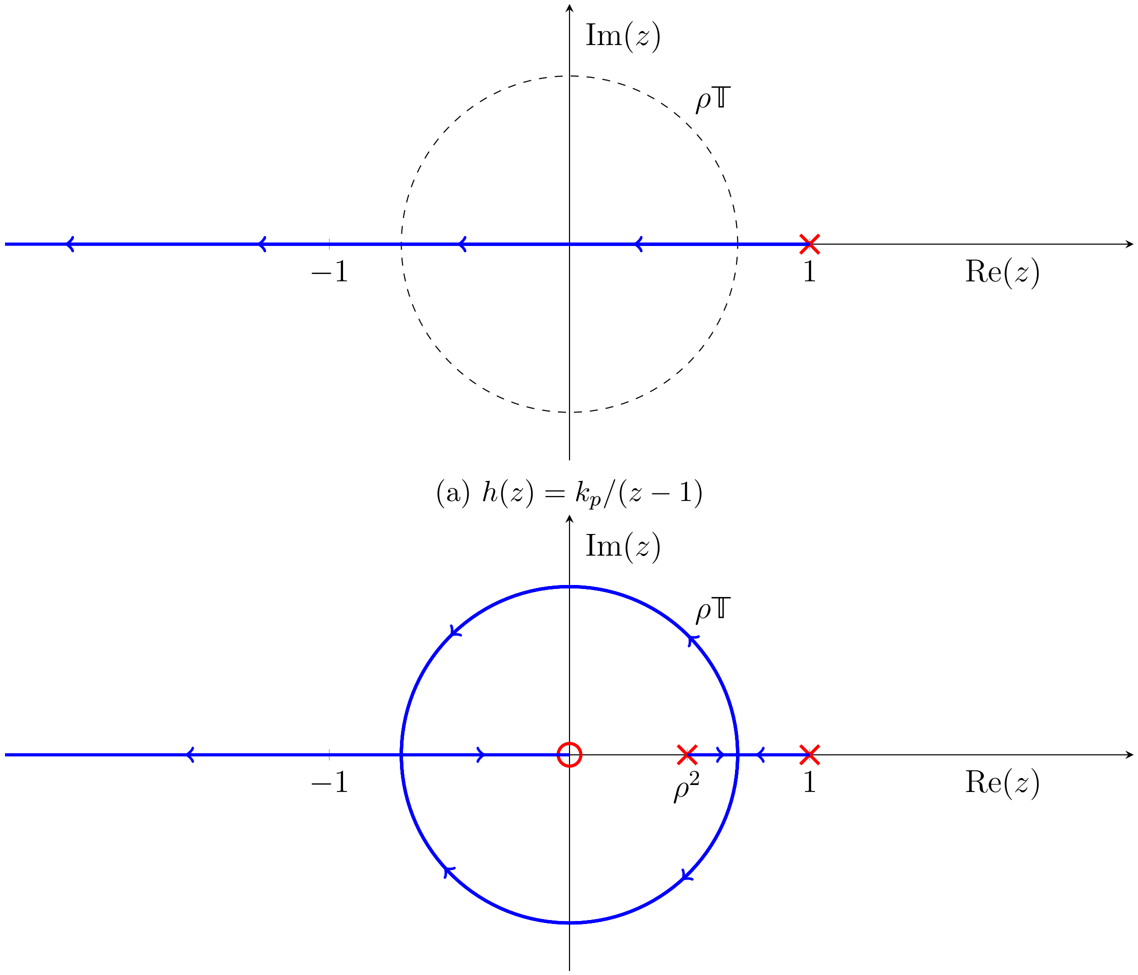
If $h(z)$ contains a model of the (non-trivial) input, then the estimator in Fig. 4.4a is internally stable but not robust to initial conditions, while the estimator in Fig. 4.4b is robust to initial conditions but not internally stable. In particular, neither estimator is capable of achieving zero steady-state error, internal stability, and robustness to initial conditions.

The estimator in Fig. 4.4c cannot achieve zero steady-state error, even if $h(z)$ contains a model of the input, since $h(z)$ is in the forward path. However, this estimator with $h(z)$ strictly stable is robust to initial conditions, internally stable, and can be cascaded to achieve arbitrarily small steady-state error for bandlimited input signals; see Chapter 5.

4.2.1. Optimizing the convergence rate

We now optimize the convergence rate while requiring that the estimator contain the model for constant inputs, i.e., $h(z)$ must contain a pole at $z = 1$. Suppose the graph is undirected, connected, and has Laplacian eigenvalues in $[\lambda_{\min}, \lambda_{\max}]$. Each estimator in Fig. 4.4 has the characteristic equation

$$0 = 1 + \lambda h(z) \tag{4.17}$$



(a) $h(z) = k_p/(z-1)$

(b) $h(z) = k_p\,z/((z-\rho^2)(z-1))$

Figure 4.5. Root locus design of the P estimator.

where $\lambda \in \text{eig}(L)$. Since the graph is undirected, λ is real and the closed-loop poles can be found using the root locus of (4.17). The simplest choice for the filter $h(z)$ is a system with one state, $h(z) = k_p/(z-\gamma)$, where we need $\gamma = 1$ in order for the estimator to be exact. To optimize the convergence rate, we want to minimize ρ such that the closed-loop poles of the system are inside $\rho\mathbb{T}$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. For $h(z) = k_p/(z-1)$ this is the same as the static estimator case, and the root locus is shown in Fig. 4.5a. However, the convergence rate can be improved when the graph is constant by introducing more dynamics. Instead of

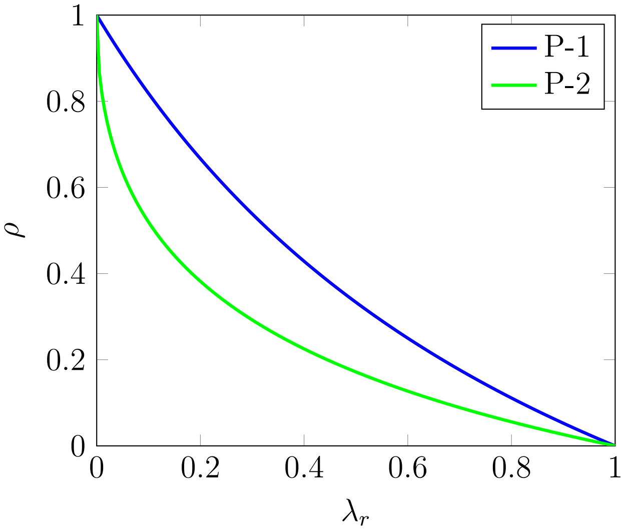


Figure 4.6. Convergence rate of the P estimator.

having the pole move straight across $\rho\mathbb{T}$ on the real axis, the closed-loop poles can be inside $\rho\mathbb{T}$ for a larger interval $[\lambda_{\min}, \lambda_{\max}]$ if we have two poles which go around the circle $\rho\mathbb{T}$. This is accomplished using

$$h(z) = \frac{k_p\, z}{(z-\rho^2)(z-1)} \tag{4.18}$$

which produces the root locus in Fig. 4.5b. The conditions used to solve for k_p and ρ similar to those from the static estimator case. We want both poles at $z = \rho$ when $\lambda = \lambda_{\min}$ and both poles at $z = -\rho$ when $\lambda = \lambda_{\max}$, i.e.,

$$0 = 1 + \lambda_{\min}\, h(\rho) \qquad \text{and} \qquad 0 = 1 + \lambda_{\max}\, h(-\rho). \tag{4.19}$$

Solving these equations for k_p and ρ gives

$$k_p = \frac{4}{(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})^2} \qquad \text{and} \qquad \rho = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}} \tag{4.20}$$

where $\lambda_r := \lambda_{\min}/\lambda_{\max}$. The extra dynamics are often referred to as adding *momentum* to the system. Note that this design only works when the graph is constant.

A simple way of producing the same result is to begin with the one-state transfer function and apply the palindromic transformation in Theorem 5. In particular, replacing z with $z + \rho^2/z$ in the transfer function $k_p/(z - (1+\rho^2))$ results in (4.18). The parameters k_p and ρ can then be found by solving

$$0 = 1 + \lambda_{\min}\left(\frac{k_p}{z-(1+\rho^2)}\right)\Big|_{z=2\rho} \qquad \text{and} \qquad 0 = 1 + \lambda_{\max}\left(\frac{k_p}{z-(1+\rho^2)}\right)\Big|_{z=-2\rho} \tag{4.21}$$

which gives the same solution as (4.20). The benefit of using the palindromic transformation to design the estimator is that $h(z)$ has degree one instead of two.

P-1. *The P-1 (proportional with one state) estimator is the estimator in Fig. 4.4 with*

$$h(z) = \frac{k_p}{z-1}$$

where

$$k_p = \frac{2}{\lambda_{\max}+\lambda_{\min}} \qquad \textit{and} \qquad \rho = \frac{\lambda_{\max}-\lambda_{\min}}{\lambda_{\max}+\lambda_{\min}}.$$

Specifically, we use the terms P-1a, P-1b, and P-1c to refer to the estimators in Fig. 4.4(a,b,c), respectively.

P-2. *The P-2 (proportional with two states) estimator is the estimator in Fig. 4.4 with*

$$h(z) = \frac{k_p\, z}{(z-1)(z-\rho^2)}$$

where

$$k_p = \frac{4}{(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})^2} \qquad \textit{and} \qquad \rho = \frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}}.$$

Specifically, we use the terms P-2a, P-2b, and P-2c to refer to the estimators in Fig. 4.4(a,b,c), respectively.

4.3. Dynamic Estimators with Two Transmission Variables (PI Estimator)

The dynamic estimators in which each agent broadcasts one scalar variable per iteration in Fig. 4.4 can track the average of time-varying inputs. None of the estimators, however, are robust to initial conditions, internally stable, and exact for constant inputs. In this section, we consider feedback estimators which transmit two state variables per iteration. The block diagram of such an estimator contains two Laplacian blocks as shown in Fig. 4.7, where both proportional and integral terms appear in the feedback loop; for this reason it is referred to as the *proportional-integral (PI) estimator* [**19**]. This makes it possible to design an estimator which has all of the properties listed in Problem 3.

From the analysis in Chapter 2, we want the filter $h_2(z)$ to have a pole at $z = 1$ so that the estimator is both internally stable and robust to initial conditions. Also, the filters $h_1(z)$ and $h_2(z)$ must be strictly proper so that the estimator can be implemented using one-hop

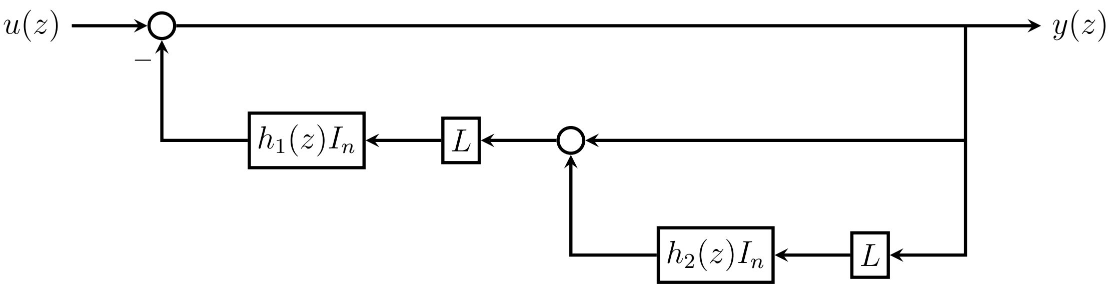


Figure 4.7. Block diagram of the proportional-integral estimator.

communication. Then we can write

$$h_1(z) = \left[\begin{array}{c|c} a_1 & b_1 \\ \hline c_1 & 0 \end{array}\right] \qquad \text{and} \qquad h_2(z) = \left[\begin{array}{c|c} a_2 & b_2 \\ \hline c_2 & 0 \end{array}\right] \tag{4.22}$$

where the dimensions are $a_1 \in \mathbb{R}^{p_1 \times p_1}$, $b_1 \in \mathbb{R}^{p_1}$, $c_1^T \in \mathbb{R}^{p_1}$, $a_2 \in \mathbb{R}^{p_2 \times p_2}$, $b_2 \in \mathbb{R}^{p_2}$, $c_2^T \in \mathbb{R}^{p_2}$. The estimator is implemented on agent i using the equations

$$\begin{aligned} \nu_{k+1}^i &= a_1\, \nu_k^i + b_1 \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(y_k^i - y_k^j) + b_1 \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(c_2 \eta_k^i - c_2 \eta_k^j) \\ \eta_{k+1}^i &= a_2\, \eta_k^i + b_2 \sum_{j \in \mathcal{N}_{\text{in}}(i)} a_{ij}(y_k^i - y_k^j) \\ y_k^i &= u_k^i - c_1\, \nu_k^i \end{aligned} \tag{4.23}$$

where $\nu_0^i \in \mathbb{R}^{p_1}$ and $\eta_0^i \in \mathbb{R}^{p_2}$ are the initial conditions on agent i, and ν_k^i and η_k^i are the internal states on agent i at time k. At time k, agent i broadcasts the two scalar variables y_k^i and $c_2 \eta_k^i$ to neighboring agents. In compact form, this can be written

$$\left[\begin{array}{c|c} A(L) & B(L) \\ \hline C(L) & D(L) \end{array}\right] = \left[\begin{array}{cc|c} a_1 & 0 & 0 \\ 0 & a_2 & 0 \\ \hline -c_1 & 0 & 1 \end{array}\right] \otimes I + \left[\begin{array}{cc|c} -b_1 c_1 & b_1 c_2 & b_1 \\ -b_2 c_1 & 0 & b_2 \\ \hline 0 & 0 & 0 \end{array}\right] \otimes L. \tag{4.24}$$

4.3.1. Convergence Rate Optimization

We now consider how to design $h_1(z)$ and $h_2(z)$ to find the minimum ρ such that the error converges linearly to zero with rate ρ when the input signals are constant. Using the separated system, we see that the convergence rate is the maximum modulus of the closed-loop poles of the separated system. The closed-loop poles of the PI estimator are the solutions to

$$0 = 1 + \lambda h_1(z)\,[1 + \lambda h_2(z)] := F(z, \lambda) \tag{4.25}$$

where $\lambda \in \operatorname{eig}(L)$. Recall that we need $h_1(z)$ and $h_2(z)$ to be strictly proper so that the estimator can be implemented in one-hop, and $h_2(z)$ must have a pole at $z = 1$ to achieve zero steady-state error for constant input signals.

Solving (4.25) is a quadratic root locus problem which has been studied in [**63**]. However, instead of viewing the problem as a quadratic root locus, we do the design as two nested linear root locus problems. For fixed $\bar{\lambda}$, the closed-loop poles of the system are on the h_1-locus

$$0 = 1 + \lambda h_1(z)\left[1 + \bar{\lambda} h_2(z)\right] \tag{h_1-locus}$$

when $\lambda = \bar{\lambda}$. To design the h_1-locus, we need the roots of

$$0 = 1 + \bar{\lambda} h_2(z) \tag{h_2-locus}$$

which is a root locus in the parameter $\bar{\lambda}$. Using the decomposition $h_2(z) = n_2(z)/d_2(z)$, we see that

$$1 + \bar{\lambda} h_2(z) = \frac{d_2(z) + \bar{\lambda} n_2(z)}{d_2(z)} = \frac{\text{closed-loop poles}}{\text{open-loop poles}},$$

so the closed-loop poles of the h_2-locus become open-loop zeros in the h_1-locus, and open-loop poles of the h_2-locus remain open-loop poles in the h_1-locus.

To design the estimator, we first choose the poles and zeros of $h_2(z)$ such that the closed-loop poles are inside $\rho\mathbb{T}$ with $\rho < 1$, keeping in mind that the closed-loop poles become the open-loop zeros of the h_1-locus. Then we choose the poles and zeros of $h_1(z)$ such that the h_1-locus remains inside $\rho\mathbb{T}$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. Conditions from locations on the root loci are used to solve for ρ and the gains of $h_1(z)$ and $h_2(z)$. Then the estimator satisfies all of the conditions in Problem 3 where ρ is the convergence rate.

We now use the developed root locus technique to design the PI estimator for two choices of $h_1(z)$ and $h_2(z)$ which have two and four internal state variables per agent, respectively.

4.3.2. Case: Two internal state variables per agent

When each agent has two internal state variables, we can parameterize $h_1(z)$ and $h_2(z)$ as

$$h_1(z) = \frac{k_p}{z-\gamma} \qquad \text{and} \qquad h_2(z) = \frac{k_I}{z-1}. \tag{4.26}$$

Since the poles of $h_1(z)$ must lie inside $\rho\mathbb{T}$, we must have $\gamma \in [-\rho, \rho]$. To design the h_2-locus, we choose the gain k_I such that the pole is at $z = \rho$ when $\bar{\lambda} = \lambda_{\min}$ as shown in Figure 4.8a. Therefore, we have

$$0 = 1 + \lambda_{\min}\, h_2(\rho) \tag{4.27}$$

which gives

$$k_I = \frac{1-\rho}{\lambda_{\min}}. \tag{4.28}$$

The h_1-locus has relative degree one, so the closed-loop pole approaches $z = -\infty$ along the negative real axis as $\lambda \to \infty$. To stabilize the pole for the largest ratio $\lambda_{\min}/\lambda_{\max}$, we choose $\gamma = \rho$. Two conditions are needed to solve for k_p and ρ. The root loci are shown for small λ_r in Figure 4.8. Due to the shape of the locus, the closed-loop poles can only exit $\rho\mathbb{T}$ at $z = -\rho$ or when the closed-loop poles are complex conjugates with magnitude ρ. To force the closed-loop poles to be on $\rho\mathbb{T}$ when $\lambda = \lambda_{\max}$, we use $c_0/c_2 = \rho^2$ from Vieta's formulas where c_j are the coefficients of $n(z, \lambda_{\max})$ with $F(z, \lambda) = n(z, \lambda)/d(z, \lambda)$, i.e.,

$$n(z, \lambda_{\max}) = \sum_{j=0}^{2} c_j \, z^j, \tag{4.29}$$

For small λ_r, we prohibit the poles from crossing the point $z = -\rho$ more than once by setting $0 = r(-\rho)$ where $r(z)$ is the discriminant of $F(z, \lambda)$ with respect to λ, i.e.,

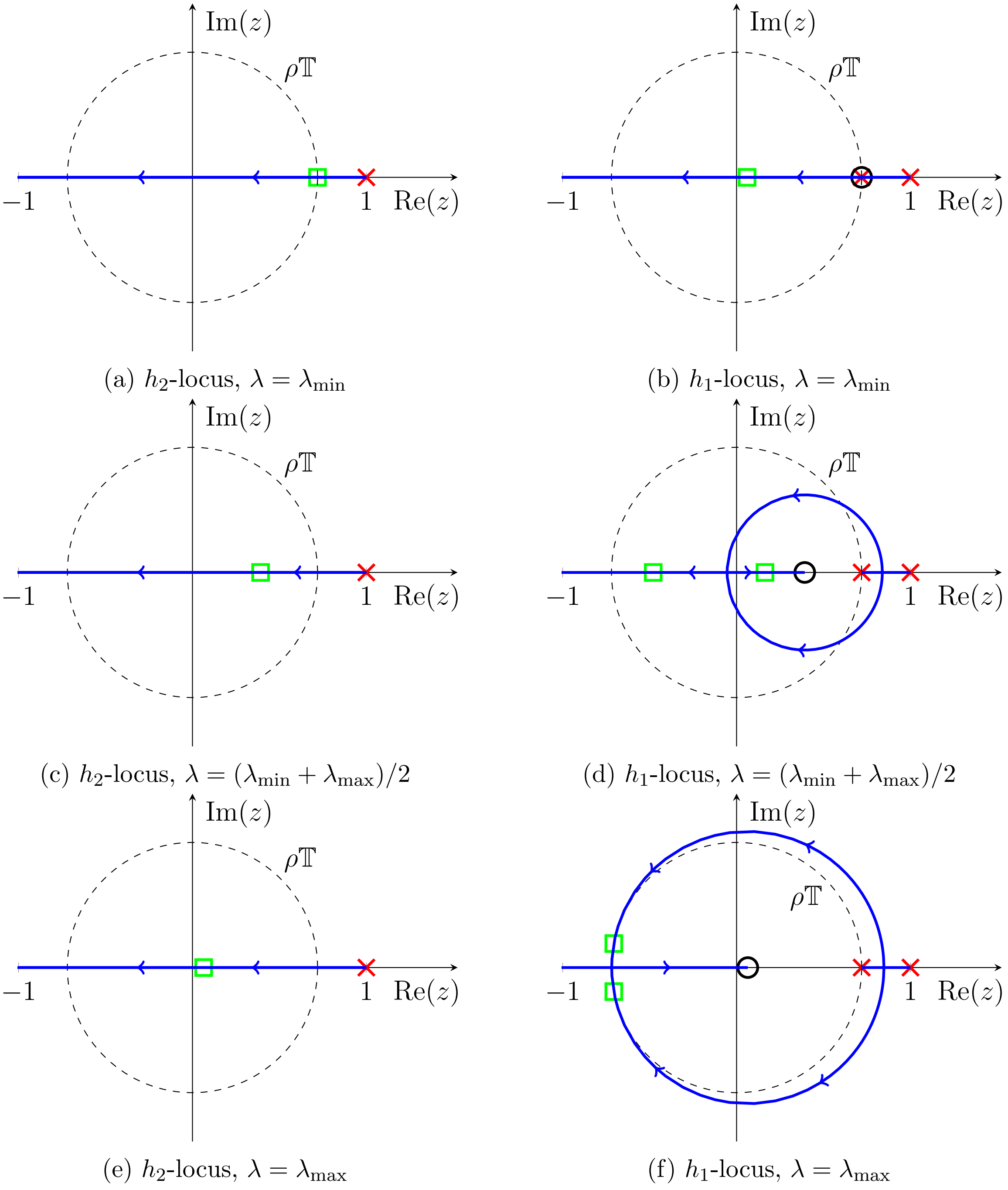
$$r(z) = h_1(z)\left[h_1(z) - 4h_2(z)\right]. \tag{4.30}$$

For large λ_r, however, the pole is outside $\rho\mathbb{T}$ for $\lambda < \lambda_{\min}$, so we force it back inside $\rho\mathbb{T}$ at $\lambda = \lambda_{\min}$ using $0 = F(-\rho, \lambda_{\min})$.

To summarize, the conditions from the root locus are

$$0 = \rho^2 - \frac{c_0}{c_2} \tag{4.31}$$

$$0 = \begin{cases} r(-\rho), & \lambda_r \text{ small} \\ F(-\rho, \lambda_{\min}), & \lambda_r \text{ large.} \end{cases} \tag{4.32}$$



(a) h_2-locus, $\lambda = \lambda_{\min}$

(b) h_1-locus, $\lambda = \lambda_{\min}$

(c) h_2-locus, $\lambda = (\lambda_{\min} + \lambda_{\max})/2$

(d) h_1-locus, $\lambda = (\lambda_{\min} + \lambda_{\max})/2$

(e) h_2-locus, $\lambda = \lambda_{\max}$

(f) h_1-locus, $\lambda = \lambda_{\max}$

Figure 4.8. Root locus design of the PI estimator with two internal state variables per agent using $\lambda_r = 0.3$ (small λ_r case). $\times$'s are open-loop poles, $\bigcirc$'s are open-loop zeros, and $\square$'s are closed-loop poles at the given value of λ.

We can solve the conditions for both the small and large λ_r cases, and then use the condition $0 = r(-\rho) = F(-\rho, \lambda_{\min})$ to find the transition point between the two solutions. Doing so produces the PI-2 estimator which is summarized below.

PI-2. *The PI-2 (proportional integral with two states) estimator is the estimator in Fig. 4.7 with*

$$h_1(z) = \frac{k_p}{z-\rho} \qquad \textit{and} \qquad h_2(z) = \frac{k_I}{z-1}$$

where the parameters are

$$k_p = \frac{1}{\lambda_{\max}} \frac{\rho(1-\rho)\lambda_r}{\rho + \lambda_r - 1} \qquad \textit{and} \qquad k_I = \frac{1-\rho}{\lambda_{\min}}$$

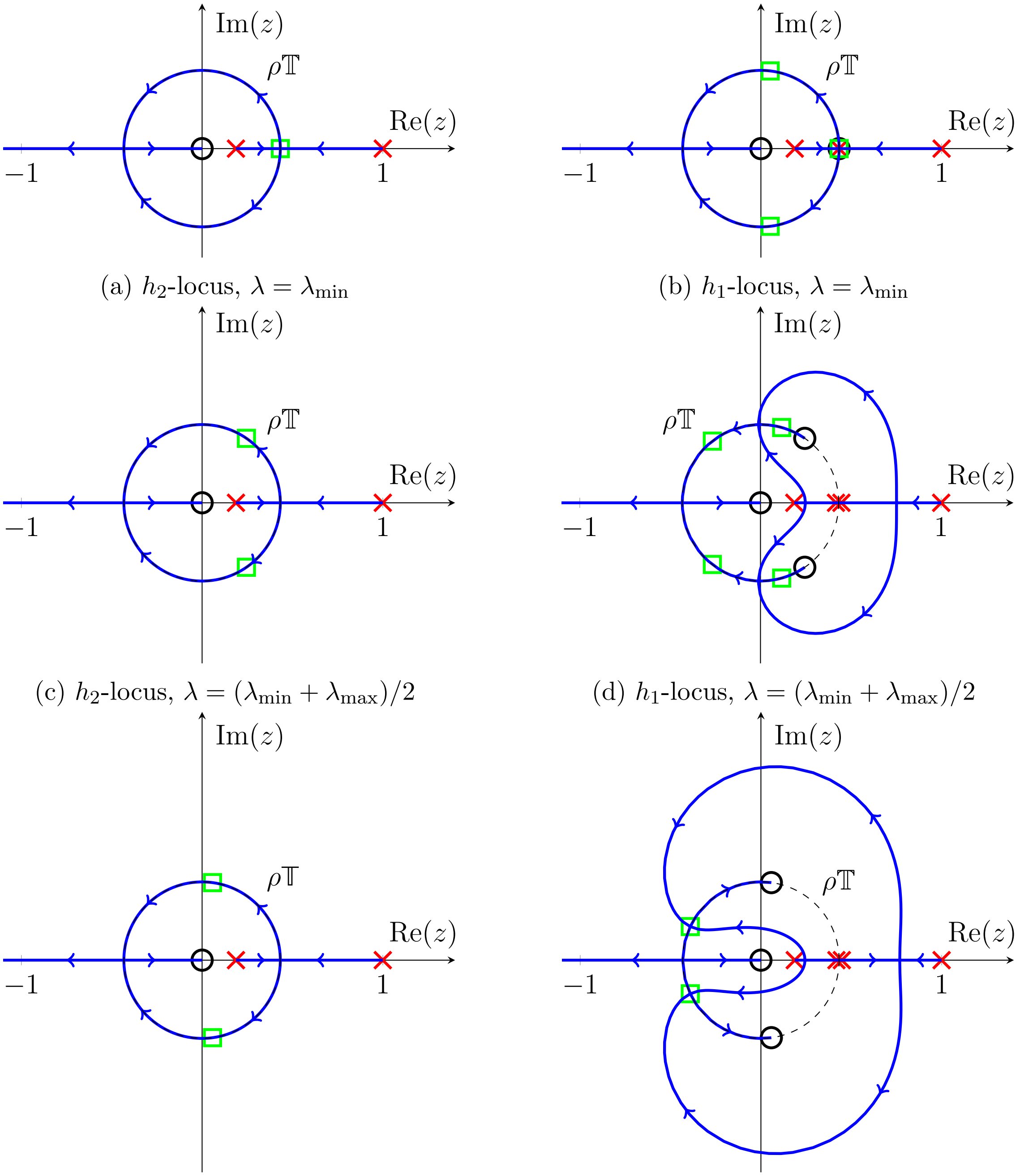
and the convergence rate is

$$\rho = \begin{cases} \dfrac{\lambda_r^2 - 8\lambda_r + 8}{8 - \lambda_r^2}, & 0 < \lambda_r \leq 3 - \sqrt{5} \\ \dfrac{\sqrt{(1-\lambda_r)(5\lambda_r^2 - \lambda_r^3 + 4)} - \lambda_r + \lambda_r^2}{2(\lambda_r^2 + 1)}, & 3 - \sqrt{5} < \lambda_r \leq 1. \end{cases}$$

The parameters γ, k_p, k_I, and ρ were optimized numerically in [**64**], and the global optimum was found. The root locus design procedure, however, gives more insight along with giving a closed-form solution. By comparing solutions, we see that the root locus design gives the same solution as that of [**64**], and is therefore the optimal set of parameters.

4.3.3. Case: Four internal state variables per agent

We now consider the case when each agent has four internal state variables. To do the analysis, we make use of the palindromic transformation in Theorem 5. In this case, we need



(a) h_2-locus, $\lambda = \lambda_{\text{min}}$

(b) h_1-locus, $\lambda = \lambda_{\text{min}}$

(c) h_2-locus, $\lambda = (\lambda_{\text{min}} + \lambda_{\text{max}})/2$

(d) h_1-locus, $\lambda = (\lambda_{\text{min}} + \lambda_{\text{max}})/2$

(e) h_2-locus, $\lambda = \lambda_{\text{max}}$

(f) h_1-locus, $\lambda = \lambda_{\text{max}}$

Figure 4.9. Root locus design of the PI estimator with four internal state variables per agent using $\lambda_r = 0.3$ (small λ_r case). $\times$'s are open-loop poles, $\bigcirc$'s are open-loop zeros, and $\square$'s are closed-loop poles at the given value of λ.

$\tilde{h}_1(z)$ to have a pole at $z = 1 + \rho^2$ and the pole of $\tilde{h}_2(z)$ must be inside $[-2\rho, 2\rho]$. Therefore, we have

$$h_1(z) = \frac{k_p\, z}{(z-\rho)^2}, \qquad\qquad h_2(z) = \frac{k_I\, z}{(z-1)(z-\rho^2)} \tag{4.33}$$

and the corresponding palindromic system is

$$\tilde{h}_1(z) = \frac{k_p}{z - 2\rho}, \qquad\qquad \tilde{h}_2(z) = \frac{k_I}{z - (1+\rho^2)} \tag{4.34}$$

The analysis is now similar to that of the two-variable case, except that we use $\tilde{h}_1$ and $\tilde{h}_2$ and we need the roots to be in the real interval $[-2\rho, 2\rho]$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. Analogous to before, we use the condition

$$0 = 1 + \lambda_{\min}\, h_2(2\rho) \tag{4.35}$$

to obtain

$$k_I = \frac{(1-\rho)^2}{\lambda_{\min}}. \tag{4.36}$$

The poles can only exist $[-2\rho, 2\rho]$ either at an endpoint or when the poles are repeated (at a breakaway point). Define

$$\tilde{F}(z, \lambda) := 1 + \lambda \tilde{h}_1(z)[1 + \lambda \tilde{h}_2(z)] \tag{4.37}$$

and the discriminant with respect to the palindromic system

$$\tilde{r}(z) = \tilde{h}_1(z)\left[\tilde{h}_1(z) - 4\tilde{h}_2(z)\right]. \tag{4.38}$$

For small λ_r, we use $0 = \tilde{r}(-2\rho)$ to force the poles not to cross $z = -2\rho$ more than once. For large λ_r, the pole leaves $[-2\rho, 2\rho]$ for $\lambda < \lambda_{\min}$ so we force it back in the interval using $0 = \tilde{F}(-2\rho, \lambda_{\min})$. Letting $\tilde{F}(z, \lambda) = \tilde{n}(z, \lambda)/\tilde{d}(z, \lambda)$ and defining

$$\tilde{n}(z, \lambda_{\max}) = \sum_{j=0}^{2} \tilde{c}_j\, z^j, \tag{4.39}$$

we force repeated poles at $\lambda = \lambda_{\max}$ by setting the discriminant to zero, i.e., $0 = c_1^2 - 4c_0c_2$.

To summarize, the conditions are

$$0 = c_1^2 - 4c_0c_2 \qquad \text{and} \qquad 0 = \begin{cases} \tilde{r}(-2\rho), & \lambda_r \text{ small} \\ \tilde{F}(-2\rho, \lambda_{\min}), & \lambda_r \text{ large.} \end{cases} \tag{4.40}$$

Solving these conditions for the estimator parameters gives the PI-4 estimator. The convergence rate of the PI estimator with both two and four internal state variables per agent is plotted in Fig. 4.10 as a function of $\lambda_r = \lambda_{\min}/\lambda_{\max}$. Note this design could also be done without the palindromic transformation, but would then require analyzing a system with degree four instead of two. The h_1 and h_2 locus are shown in Fig. 4.9. Since the system is palindromic, the closed-loop poles are always on $\rho\mathbb{T}$.

The results for the PI-2 and PI-4 estimators are summarized in the following theorem.

Theorem 11 (PI estimator)**.** *Suppose $\mathcal{G}$ is a constant, connected, and undirected graph whose Laplacian matrix has nonzero eigenvalues in the interval $[\lambda_{\min}, \lambda_{\max}]$, and suppose the input signals are constant. Then the PI-2 and PI-4 estimators satisfy all of the conditions in Problem 3, i.e., the error converges to zero linearly with rate ρ from any initial state.*

PI-4. *The PI-4 (proportional-integral with four states) estimator is the estimator in Fig. 4.7 with*

$$h_1(z) = \frac{k_p\, z}{(z-\rho)^2} \qquad \textit{and} \qquad h_2(z) = \frac{k_I\, z}{(z-1)(z-\rho^2)}$$

where the parameters are

$$k_p = \frac{(1-\rho)^2}{\lambda_{\min}}\Big(2-\lambda_r+2\sqrt{1-\lambda_r}\Big) \qquad \textit{and} \qquad k_I = \frac{(1-\rho)^2}{\lambda_{\min}}$$

and the convergence rate is

$$\rho = \begin{cases} \dfrac{\lambda_r+6-2\sqrt{1-\lambda_r}-4\sqrt{2+\lambda_r-2\sqrt{1-\lambda_r}}}{2-\lambda_r+2\sqrt{1-\lambda_r}}, & 0<\lambda_r\le 2(\sqrt{2}-1) \\[2ex] \dfrac{\lambda_r-1+2\sqrt{1-\lambda_r}}{3+\lambda_r}, & 2(\sqrt{\lambda_r}-1)<\lambda_r\le 1. \end{cases}$$

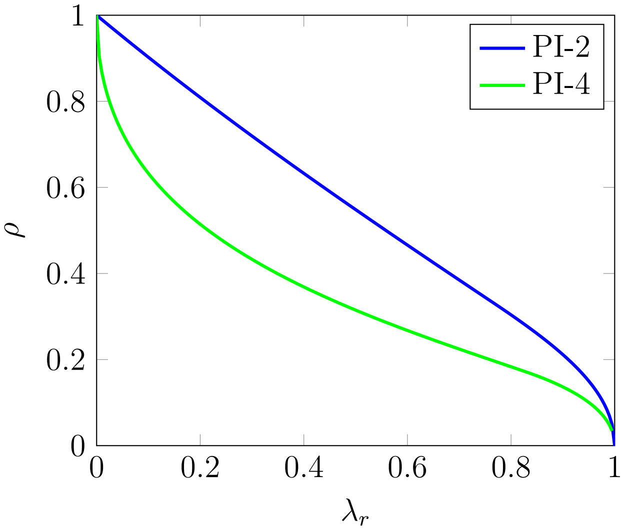


Figure 4.10. Convergence rate of the PI estimator.

4.4. Polynomial Filter Estimator

The previous feedback designs have all used one-hop communication. We designed more complex estimators (such as the PI estimator) by introducing more dynamics on each agent. In this section we design feedback estimators with simple dynamics (one or two states per agent), but allow for an arbitrary number of communication hops. Since multiple rounds of communication are required to obtain the output, we normalize the convergence rate by the number of communication hops. For an r-hop estimator with convergence rate ρ, the *normalized convergence rate* is defined as $\tilde{\rho} := \sqrt[r]{\rho}$. We are able to show the following for estimators with one or two internal states per agent:

- For estimators which are not robust to initial conditions, the optimal normalized convergence rate is obtained by an estimator with two internal states per agent using one-hop communication. This is equivalent to the P-2a estimator which has convergence rate $\rho = (1 - \sqrt{\lambda_r})/(1 + \sqrt{\lambda_r})$.
- For estimators which are robust to initial conditions, the optimal normalized convergence rate is the same as the previous case, but is only achieved (using either one or two internal states per agent) in the limit as the number of communication hops approaches infinity.

The design of multi-hop estimators uses a polynomial filter to shape the spectrum of the Laplacian matrix [**28**], and are therefore called *polynomial filter estimators*. These estimators apply a polynomial τ_r of degree r to the Laplacian to optimize the convergence rate. For example, a one-state static polynomial filter estimator is given by

$$x_{k+1} = \tau_r(L)\, x_k, \quad x_0 = u \tag{4.41}$$

where

$$\tau_r(L) = I + \sum_{i=1}^{r} L^i \prod_{j=1}^{i} \tau_j = I + \tau_1 L\Big(I + \tau_2 L\big[I + \tau_3 L(\ldots)\big]\Big) \tag{4.42}$$

is a polynomial of degree r in the matrix L. A specific example is the static estimator in Fig. 4.1 which uses the choice $\tau_1(L) = I - k_p L$. For the separated system, τ_r is a polynomial in the scalar $\lambda \in \mathrm{eig}(L)$ given by

$$\tau_r(\lambda) = 1 + \sum_{i=1}^{r} \lambda^i \prod_{j=1}^{i} \tau_j = 1 + \tau_1 \lambda\Big(1 + \tau_2 \lambda\big[1 + \tau_3 \lambda(\ldots)\big]\Big). \tag{4.43}$$

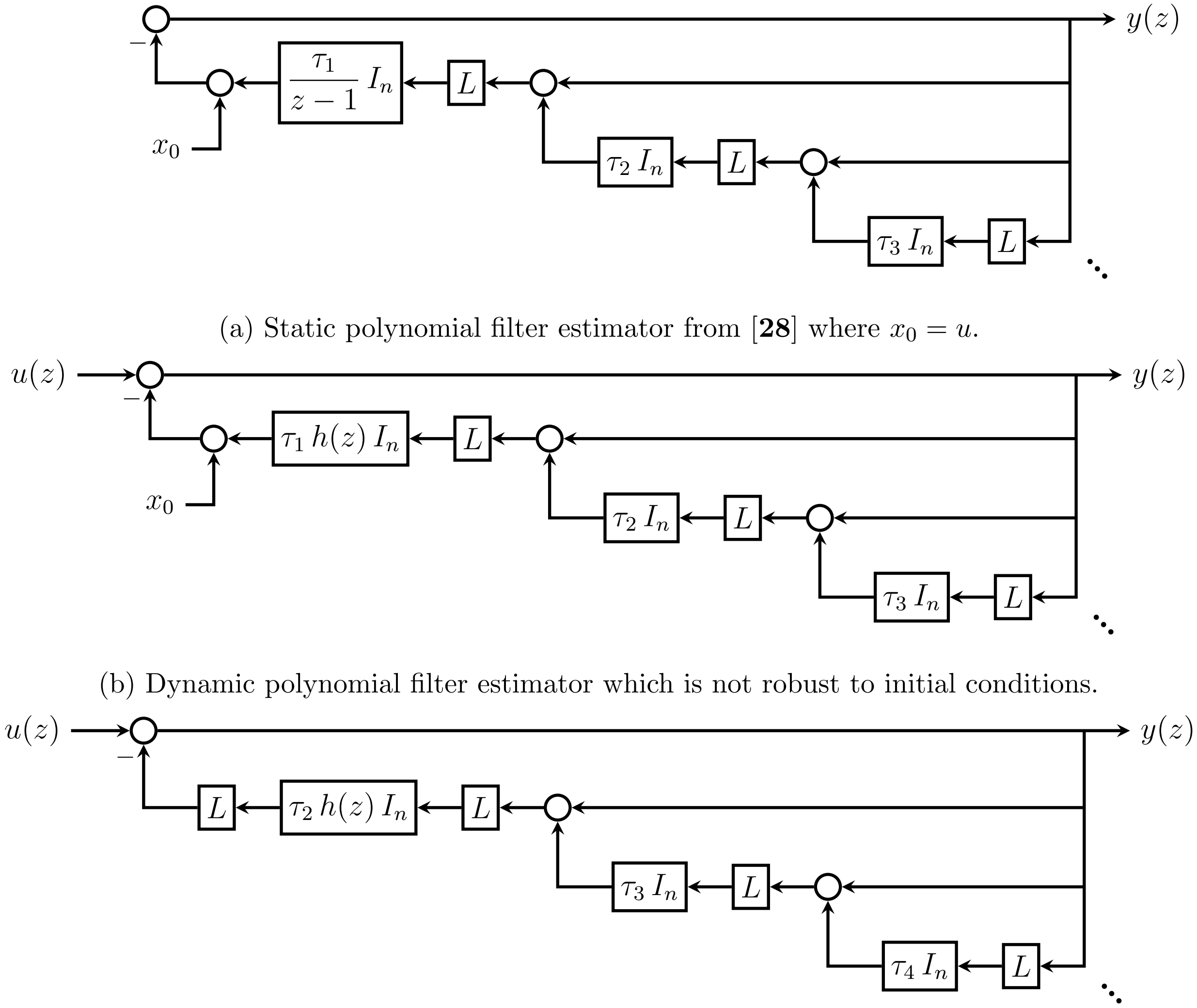
The characteristic equation of the separated system is

$$0 = z - \tau_r(\lambda) \tag{4.44}$$

which has a single root at $\tau_r(\lambda)$. The convergence rate can then be optimized by designing τ_r such that $|\tau_r(\lambda)| \leq \rho$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

The static polynomial filter is not capable of tracking time-varying input signals, but the estimator in Fig. 4.11b with $h(z) = 1/(z-1)$ is dynamic and has the same characteristic polynomial (and therefore same convergence rate). The estimator is not robust to initial conditions since the integrator state does not pass through the Laplacian before reaching the output. To make the estimator robust to initial conditions, we add a Laplacian block after the integrator as shown in Fig. 4.11c.

The number of states variables on each agent corresponds to the degree of the transfer function $h(z)$. Similar to the P-1 and P-2 estimators, we consider the case of one and two variables with $h_1(z) = 1/(z-1)$ and $h_2(z) = z/((z-\rho^2)(z-1))$, respectively. The



(a) Static polynomial filter estimator from [**28**] where $x_0 = u$.

(b) Dynamic polynomial filter estimator which is not robust to initial conditions.

(c) Dynamic polynomial filter estimator which is robust to initial conditions.

Figure 4.11. Block diagrams of polynomial filter estimators.

characteristic equation of the separated system is

$$0 = 1 + h(z)\,\tau(\lambda) \tag{4.45}$$

where $\lambda \in \operatorname{eig}(L)$. For the different choices of $h(z)$, the characteristic polynomial is

$$F(z,\lambda) = \begin{cases} z - [1 - \tau(\lambda)], & h(z) = \frac{1}{z-1} \\ z^2 - [1 + \rho^2 - \tau(\lambda)]\, z + \rho^2, & h(z) = \frac{z}{(z-\rho^2)(z-1)}. \end{cases} \tag{4.46}$$

For the estimator to be exact, we need $h(z)$ to have a pole at $z = 1$ and $\tau(0) = 0$. To be robust to initial conditions, λ^2 must factor out of $\tau(\lambda)$ so that $h(z)$ can be between the two Laplacian blocks as in Fig. 4.11(c). This is the case if $\tau'(0) = 0$.

The design of the polynomial filter estimator can then be stated as follows.

Problem 4 (Polynomial filter design)**.** *Given lower and upper bounds on the nonzero Laplacian eigenvalues $\lambda_{\min}$ and $\lambda_{\max}$ with $0 < \lambda_{\min} \leq \lambda_{\max}$, solve*

$$\tilde{\rho} = \min_{r,\ \tau} \max_{\substack{z \in \mathbb{C} \\ \lambda \in [\lambda_{\min}, \lambda_{\max}]}} \sqrt[r]{|z|} \qquad s.t. \quad 0 = F(z,\lambda) \tag{4.47}$$

where $F(z,\lambda)$ is given by (4.46)*, and τ is a polynomial of degree r that satisfies the following conditions for the given estimator properties:*

$$\begin{aligned} &\textit{Exact:} && 0 = \tau(0) \\ &\textit{Robust to ICs:} && 0 = \tau'(0). \end{aligned}$$

We first consider one- and two-state polynomial filter estimators which are exact but not robust to initial conditions, and then show how to modify the design to achieve the extra robustness condition.

4.4.1. Non-robust one-state polynomial filter estimator

The non-robust polynomial filter estimator with one state per agent is shown in Fig. 4.11b with $h(z) = 1/(z-1)$. The characteristic polynomial is given by (4.46). In this case, $F(z, \lambda)$ has a single root at $1 - \tau(\lambda)$ so we want $1 - \tau(\lambda)$ to have minimum absolute value on the interval $[\lambda_{\min}, \lambda_{\max}]$ as shown in Fig. 4.12. Chebyshev polynomials of the first-kind are known to have the minimax property meaning that they have the smallest maximum absolute value on the interval $[-1, 1]$ (when normalized so that the leading coefficient is unity). As shown in [**29**], the polynomial τ which minimizes the convergence rate ρ is obtained using Chebyshev polynomials. In particular, we have

$$\tau(\lambda) = 1 - \rho\, T_r\left(\frac{2}{\lambda_{\max} - \lambda_{\min}}\lambda - \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right) \tag{4.48}$$

where $T_r(\cdot)$ is the r^{th} Chebyshev polynomial of the first-kind. We then choose ρ such that $\tau(0) = 0$ so that the estimator is exact.

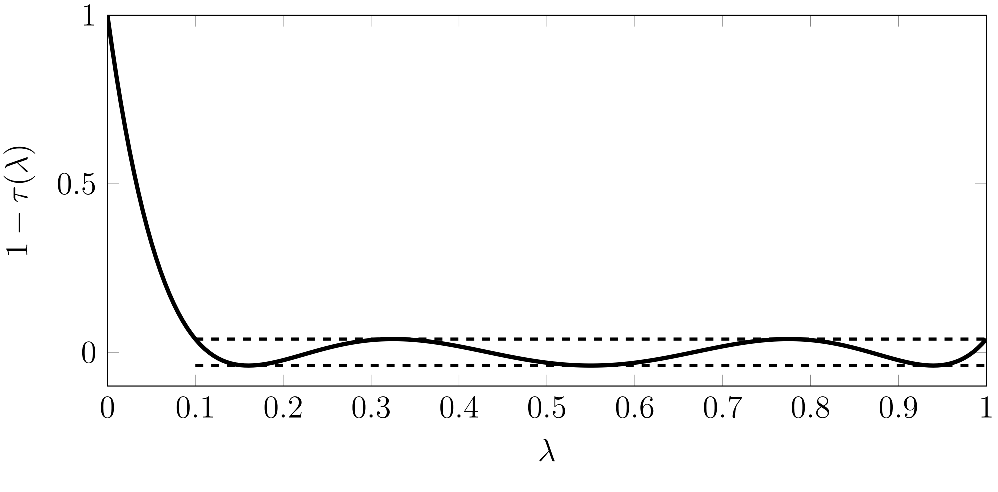


Figure 4.12. Closed-loop pole of the non-robust one-state polynomial filter estimator using $r = 6$, $\lambda_{\min} = 0.1$, and $\lambda_{\max} = 1$. The horizontal dashed lines are at $\pm\rho$. Note that the closed-loop pole has magnitude no larger than ρ for λ in the interval $[\lambda_{\min}, \lambda_{\max}]$.

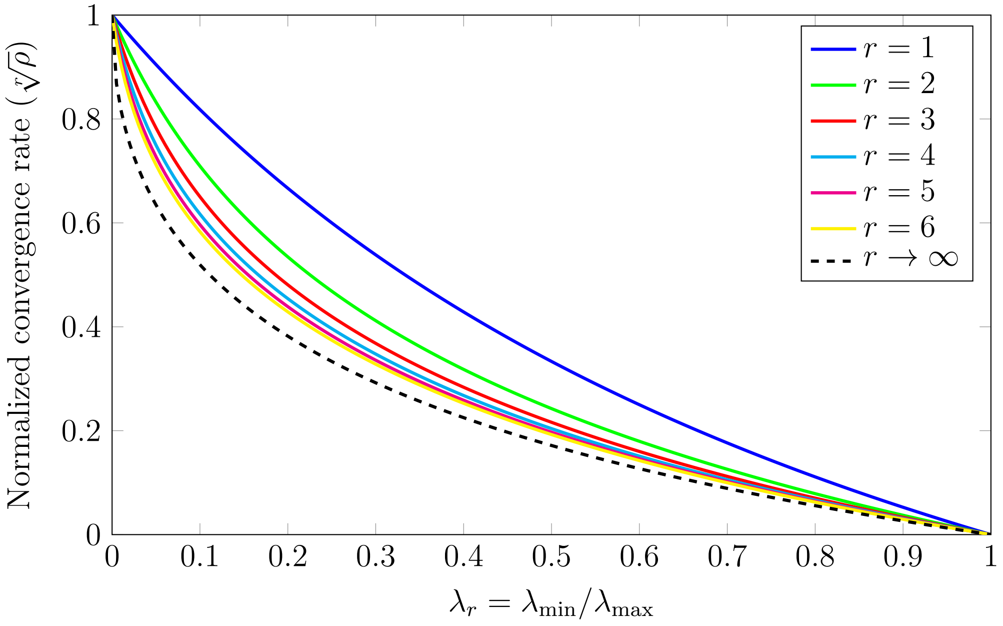


Figure 4.13. Normalized convergence rate of the PF-NR-1-r estimator. Theorem 12 proves that the convergence rate approaches the dashed black line in the limit as $r \to \infty$.

PF-NR-1-r. *The PF-NR-1-r (non-robust, one-state, r-hop polynomial filter) estimator is the estimator in Fig. 4.11b with*

$$h(z) = \frac{1}{z-1} \qquad \textit{and} \qquad \tau(\lambda) = 1 - \frac{T_r(\alpha\lambda - \beta)}{T_r(-\beta)}$$

where

$$\alpha = \frac{2}{\lambda_{\max} - \lambda_{\min}} \qquad \textit{and} \qquad \beta = \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}.$$

The convergence rate is

$$\rho = \frac{(-1)^r}{T_r\,(-\beta)}.$$

The normalized convergence rate of the PF-NR-1-r estimator is plotted in Fig. 4.13 for several values of r. In the limit as $r \to \infty$, the normalized convergence rate is given by the following theorem.

Theorem 12. *The PF-NR-1-r estimator is exact for constant input signals and internally stable, but not robust to initial conditions. Furthermore, the limit of the normalized convergence rate as the number of communication hops approaches infinity is*

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}. \tag{4.49}$$

Before proving Theorem 12, we need the following lemma.

Lemma 6. *For $x \geq 0$, $\lim_{r\to\infty} \sqrt[r]{\cosh(rx)} = e^x$.*

Proof. Since $x \geq 0$, we can upper bound $\cosh(rx)$ by

$$\cosh(rx) = \frac{e^{rx}+e^{-rx}}{2} \leq \frac{e^{rx}+e^{rx}}{2} = e^{rx}.$$

Since $e^{-rx} > 0$, a lower bound is $\frac{1}{2}e^{rx} \leq \cosh(rx)$. Using the squeeze theorem on the inequalities

$$\frac{1}{\sqrt[r]{2}}e^x \leq \sqrt[r]{\cosh(rx)} \leq e^x$$

gives that $\lim_{r\to\infty} \sqrt[r]{\cosh(rx)} = e^x$. □

We now give the proof for Theorem 12.

Proof (Theorem 12): The estimator properties come directly from applying Thm. 4 to the estimator block diagram. Define $\beta = (\lambda_{\max}+\lambda_{\min})/(\lambda_{\max}-\lambda_{\min})$. Using the fact that

$T_r(x) = (-1)^r \cosh(r \cosh^{-1}(x))$ for $x \leq -1$,

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1}{\lim_{r\to\infty} \sqrt[r]{(-1)^r T_r(-\beta)}} = \frac{1}{\lim_{r\to\infty} \sqrt[r]{\cosh\left(r \cosh^{-1}(\beta)\right)}}.$$

Note that $\beta \geq 1$ since $0 < \lambda_{\min} \leq \lambda_{\max}$, so $\cosh^{-1}(\beta) \geq 0$. Then we can apply Lemma 6 to obtain

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1}{\exp\{\cosh^{-1}(\beta)\}}.$$

Using $\cosh^{-1}(x) = \ln[x + \sqrt{x^2 - 1}]$ and simplifying gives

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1}{\exp\{\ln[\beta + \sqrt{\beta^2 - 1}]\}} = \frac{1}{\beta + \sqrt{\beta^2 - 1}} = \frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}} = \frac{1 - \sqrt{\lambda_r}}{1 + \sqrt{\lambda_r}}$$

where $\lambda_r = \lambda_{\min}/\lambda_{\max}$. □

Note that for $r = 1$, we have

$$\sqrt[r]{\rho} = \frac{1 - \lambda_r}{1 + \lambda_r} \tag{4.50}$$

which is the same as that of the optimal estimator found in [**12**] and is equivalent to the P-1a estimator. By increasing the number of communication hops, not only does the convergence rate ρ improve, but it also improves when normalized for the extra communication. Therefore, using more communication hops improves the convergence rate for the PF-NR-1-r (non-robust, one-state, r-hop polynomial filter) estimator.

4.4.2. Non-robust two-state polynomial filter estimator

The non-robust polynomial filter estimator with two states per agent is shown in Fig. 4.11b with $h(z) = z/((z-\rho^2)(z-1))$. The characteristic polynomial of the separated system is

$$F(z,\lambda) = z^2 - [1+\rho^2-\tau(\lambda)]\,z + \rho^2. \tag{4.51}$$

It can be shown that the roots of $F(z,\lambda)$ are inside $\rho\mathbb{T}$ if and only if $|1+\rho^2-\tau(\lambda)| \leq 2\rho$ (see [**65**, Lemma 1] for details). Similar to the one-state estimator, the polynomial τ which minimize ρ is given by a shifted and scaled Chebyshev polynomial,

$$\tau(\lambda) = 1+\rho^2-2\rho\,T_r\left(\frac{2}{\lambda_{\max}-\lambda_{\min}}\lambda - \frac{\lambda_{\max}+\lambda_{\min}}{\lambda_{\max}-\lambda_{\min}}\right). \tag{4.52}$$

Once again, we choose ρ such that $0 = \tau(0)$ so that the estimator is exact.

PF-NR-2-r. *The PF-NR-2-r (non-robust, two-state, r-hop polynomial filter) estimator is the estimator in Fig. 4.11b with*

$$h(z) = \frac{z}{(z-\rho^2)(z-1)} \qquad \textit{and} \qquad \tau(\lambda) = 1+\rho^2-2\rho\,T_r(\alpha\lambda-\beta)$$

where

$$\alpha = \frac{2}{\lambda_{\max}-\lambda_{\min}} \qquad \textit{and} \qquad \beta = \frac{\lambda_{\max}+\lambda_{\min}}{\lambda_{\max}-\lambda_{\min}}.$$

The convergence rate is

$$\rho = (-1)^r\,T_r(-\beta) - \sqrt{T_r^2(-\beta)-1}.$$

Theorem 13. *The PF-NR-2-r estimator is exact for constant input signals and internally stable, but not robust to initial conditions. Furthermore, the normalized convergence rate is*

$$\sqrt[r]{\rho} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}} \tag{4.53}$$

for all $r \geq 1$.

Proof. Similar to the proof of Theorem 12, define $\beta = (1+\lambda_r)/(1-\lambda_r)$ and use the fact that $T_r(x) = (-1)^r \cosh(r \cosh^{-1}(x))$ for $x \leq -1$ to write

$$\rho = (-1)^r T_r(-\beta) - \sqrt{T_r^2(-\beta) - 1} = \cosh(r\cosh^{-1}(\beta)) - \sqrt{\cosh^2(r\cosh^{-1}(\beta)) - 1}.$$

Using the common identities $\cosh^2(x) - 1 = \sinh^2(x)$ and $\cosh(x) - \sinh(x) = e^{-x}$ gives

$$\rho = \cosh(r\cosh^{-1}(\beta)) - \sinh(r\cosh^{-1}(\beta)) = \exp\{-r\cosh^{-1}(\beta)\}.$$

Once again using $\cosh^{-1}(x) = \ln[x + \sqrt{x^2-1}]$ and simplifying gives

$$\begin{aligned} \rho &= \exp\{-r\ln[\beta + \sqrt{\beta^2-1}]\} = \exp\{\ln[(\beta + \sqrt{\beta^2-1}]^{-r}\} \\ &= \frac{1}{(\beta + \sqrt{\beta^2-1})^r} = \left(\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}\right)^r . \end{aligned}$$

□

Note that the performance of the two-state estimator is always better than that of the one-state estimator, and the performance of the one-state estimator approaches that of the two-state estimator as the number of communication hops approaches infinity.

Since it does not benefit the two-state estimator to use any additional communication hops, the optimal use of memory in the non-robust case is to use the one-hop two-state estimator, i.e., the PF-NR-2-1 estimator. Note that this is equivalent to the two-state proportional estimator in Fig. 4.4(a), i.e., the P-2a estimator.

So far we have only considered polynomial filter estimators which are not robust to initial conditions. We now analyze the polynomial filter estimator in Fig. 4.11c which is robust to initial conditions.

4.4.3. Robust one-state polynomial filter estimator

For robust polynomial filter estimators, the characteristic polynomial is still given by (4.46), although we now have the extra robustness condition $\tau'(0) = 0$. For low degree polynomials, we can solve for the optimal solution. For example, when $\tau(\lambda)$ is degree two (the lowest possible degree for robustness to initial conditions), we have $\tau(\lambda) = 1+\tau_2\lambda^2$ with $\tau(\lambda_{\min}) = \rho$ and $\tau(\lambda_{\max}) = -\rho$ which implies that

$$\tau_2 = -\frac{2}{\lambda_{\max}^2 + \lambda_{\min}^2} \qquad \text{and} \qquad \rho = \frac{1-\lambda_r^2}{1+\lambda_r^2}. \tag{4.54}$$

This convergence rate is quite slow. It can be improved using a higher degree polynomial, but this design process becomes infeasible. In the non-robust case, the optimal solution was given by Chebyshev polynomials. These no longer work in the robust case since they do not satisfy the robustness condition $\tau'(0) = 0$. To handle the extra condition, we multiply a Chebyshev polynomial by a linear term and use the extra degrees of freedom from the linear term to satisfy the robustness condition. This allows us to analyze the performance of high-degree estimators (although it is not optimal for low-degree estimators). Therefore, we let $\tau(\lambda)$ be given by

$$\tau(\lambda) = 1 - (a\,\lambda - b)\,T_{r-1}\left(\frac{2}{\lambda_{\max} - \lambda_{\min}}\lambda - \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}\right). \tag{4.55}$$

We then choose a and b such that $\tau(0) = 0$ and $\tau'(0) = 0$ so that the estimator is exact and robust to initial conditions.

PF-R-1-r. *The PF-R-1-r (robust, one-state, r-hop polynomial filter) estimator is the estimator in Fig. 4.11c with*

$$h(z) = \frac{1}{z-1} \qquad \textit{and} \qquad \tau(\lambda) = 1 - \left(1 - \alpha\lambda\,\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right)\frac{T_{r-1}(\alpha\lambda - \beta)}{T_{r-1}(-\beta)}$$

where

$$\alpha = \frac{2}{\lambda_{\max} - \lambda_{\min}} \qquad \textit{and} \qquad \beta = \frac{\lambda_{\max} + \lambda_{\min}}{\lambda_{\max} - \lambda_{\min}}.$$

The convergence rate is

$$\rho = \frac{(-1)^{r-1}}{T_{r-1}(-\beta)}\left(1 - \frac{2}{1-\lambda_r}\,\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right)$$

Theorem 14. *The PF-R-1-r estimator is exact for constant input signals, internally stable, and robust to initial conditions. Furthermore, the limit of the normalized convergence rate as the number of communication hops approaches infinity is*

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1 - \sqrt{\lambda_r}}{1 + \sqrt{\lambda_r}}. \tag{4.56}$$

Before proving Theorem 14, we need the following lemma.

Lemma 7. *For $x > 1$ and any $\gamma > 0$,*

$$\lim_{r\to\infty} \sqrt[r]{1 - \gamma\,\frac{T_r'(x)}{T_r(x)}} = 1. \tag{4.57}$$

Proof. Let $c = \sqrt{x^2-1}$ and $a = x + c$. Then $1/a = x - c$ and $|a| < 1$. Then we can write $T_r(x)$ and $T_r'(x)$ as

$$T_r(x) = \frac{a^r + a^{-r}}{2} \qquad \text{and} \qquad T_r'(x) = r\,\frac{a^r - a^{-r}}{2c}.$$

Taking the logarithm of the left-hand side of (4.57), we have

$$\lim_{r\to\infty} \log \sqrt[r]{1 - \gamma\,\frac{T_r'(x)}{T_r(x)}} = \lim_{r\to\infty} \frac{\log\left[1 - \gamma\,\frac{T_r'(x)}{T_r(x)}\right]}{r} = \lim_{r\to\infty} \frac{\log\left[1 - \gamma\,\frac{r}{c}\,\frac{a^r - a^{-r}}{a^r + a^{-r}}\right]}{r}.$$

Using L'Hôpital's rule, this gives

$$\lim_{r\to\infty} \frac{\gamma\,\frac{1 - a^{4r} - 4ra^{2r}\log(a)}{c(1+a^{2r})^2}}{1 - \gamma\,\frac{r}{c}\,\frac{a^r - a^{-r}}{a^r + a^{-r}}} = \lim_{r\to\infty} \frac{-\gamma/c}{-r\,\gamma/c} = \lim_{r\to\infty} 1/r = 0.$$

Since $\log(0) = 1$ and the $\log(\cdot)$ function is continuous, this gives the result. □

Proof (Theorem 14): The estimator is exact since $h(z)$ contains a pole at $z = 1$ and $\tau(0) = 0$. To check robustness to initial conditions, we first compute

$$\tau'(\lambda) = \alpha\frac{T_{r-1}'(-\beta)}{T_{r-1}(-\beta)}\frac{T_{r-1}(\alpha\,\lambda - \beta)}{T_{r-1}(-\beta)} - \left(1 - \alpha\,\frac{T_{r-1}'(-\beta)}{T_{r-1}(-\beta)}\,\lambda\right)\,\alpha\,\frac{T_{r-1}'(\alpha\,\lambda - \beta)}{T_{r-1}(-\beta)}$$

from which we can see that $\tau'(0) = 0$, so the estimator is robust to initial conditions.

The convergence rate is given by

$$\rho = |1 - \tau(\lambda_{\max})| = (-1)^{r-1}\left[1 - \tau(\lambda_{\max})\right] = \frac{(-1)^{r-1}}{T_{r-1}(-\beta)}\left(1 - \frac{2}{1-\lambda_r}\,\frac{T_{r-1}'(-\beta)}{T_{r-1}(-\beta)}\right).$$

Splitting the limit to be evaluated, we have

$$\begin{aligned}\lim_{r\to\infty}\sqrt[r]{\rho} &= \lim_{r\to\infty}\sqrt[r]{\frac{(-1)^{r-1}}{T_{r-1}(-\beta)}\left[1-\frac{2}{1-\lambda_r}\,\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right]}\\ &= \lim_{r\to\infty}\left(\sqrt[r-1]{\frac{(-1)^{r-1}}{T_{r-1}(-\beta)}}\right)^{\frac{r-1}{r}}\left(\sqrt[r-1]{\left[1-\frac{2}{1-\lambda_r}\,\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right]}\right)^{\frac{r-1}{r}}.\end{aligned}$$

Theorem 12 provides the solution to the limit of the first expression while Lemma 7 gives that the limit of the second expression is equal to one, so

$$\lim_{r\to\infty}\sqrt[r]{\rho} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}. \qquad \square$$

4.4.4. Robust two-state polynomial filter estimator

The design of the two-state robust estimator is similar to that of the two-state non-robust estimator except for the additional robustness condition, $\tau(0)=0$, where the characteristic polynomial is

$$F(z,\lambda) = z^2 - \left[1+\rho^2-\tau(\lambda)\right]z + \rho^2. \tag{4.58}$$

Similar to the previous cases, we choose

$$\tau(\lambda) = (1+\rho^2)\left[1-(a\lambda-b)\,T_{r-1}\left(\frac{2}{\lambda_{\max}-\lambda_{\min}}\lambda - \frac{\lambda_{\max}+\lambda_{\min}}{\lambda_{\max}-\lambda_{\min}}\right)\right]. \tag{4.59}$$

The conditions for the estimator to be exact and robust to initial conditions are the same as the robust one-state case,

$$a = -\alpha\frac{T'_{r-1}(-\beta)}{T^2_{r-1}(-\beta)} \qquad \text{and} \qquad b = -\frac{1}{T_{r-1}(-\beta)}. \tag{4.60}$$

The asymptotic convergence rate ρ is then chosen so that $|1+\rho^2-\tau(\lambda)| \leq 2\rho$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$.

PF-R-2-r. *The PF-R-2-r (robust, two-state, r-hop polynomial filter) estimator is the estimator in Fig. 4.11c with*

$$h(z) = \frac{z}{(z-\rho^2)(z-1)} \quad \textit{and} \quad \tau(\lambda) = (1+\rho^2)\left[1 - \left(1 - \alpha\lambda \frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right) \frac{T_{r-1}(\alpha\lambda-\beta)}{T_{r-1}(-\beta)}\right]$$

where

$$\alpha = \frac{2}{\lambda_{\max}-\lambda_{\min}} \qquad \textit{and} \qquad \beta = \frac{\lambda_{\max}+\lambda_{\min}}{\lambda_{\max}-\lambda_{\min}}.$$

The convergence rate is $\rho = (1-\sqrt{1-\rho_1^2})/\rho_1$ *where*

$$\rho_1 = \frac{(-1)^{r-1}}{T_{r-1}(-\beta)}\left(1 - \frac{2}{1-\lambda_r}\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right)$$

Theorem 15. *The PF-R-2-r estimator is exact for constant input signals, internally stable, and robust to initial conditions. Furthermore, the limit of the normalized convergence rate as the number of communication hops approaches infinity is*

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}. \tag{4.61}$$

Proof. The convergence rate of the PF-R-2-r estimator is the minimum ρ such that

$$2\rho \geq |(1+\rho^2)-\tau(\lambda)| = \left|(1+\rho^2)\left(1-\alpha\lambda\frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right)\frac{T_{r-1}(\alpha\lambda-\beta)}{T_{r-1}(-\beta)}\right|$$

for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. From the results for the PF-R-1-r estimator, we have that

$$\left| \left(1 - \alpha\lambda \, \frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right) \frac{T_{r-1}(\alpha\lambda - \beta)}{T_{r-1}(-\beta)} \right| \le \rho_1$$

for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$ where

$$\rho_1 = \frac{(-1)^{r-1}}{T_{r-1}(-\beta)} \left(1 - \frac{2}{1-\lambda_r} \, \frac{T'_{r-1}(-\beta)}{T_{r-1}(-\beta)}\right) \qquad \text{and} \qquad \lim_{r\to\infty} \sqrt[r]{\rho_1} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}.$$

Comparing the two bounds, we have $2\rho/(1+\rho^2) = \rho_1$ which gives

$$\rho = \frac{1 - \sqrt{1-\rho_1^2}}{\rho_1}.$$

Taking the limit, we have

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \lim_{r\to\infty} \sqrt[r]{\frac{1-\sqrt{1-\rho_1^2}}{\rho_1}} = \lim_{r\to\infty} \sqrt[r]{\rho_1} \, \lim_{r\to\infty} \sqrt[r]{\frac{1}{\rho_1^2} - \sqrt{\frac{1}{\rho_1^2} - 1}}.$$

The second limit is one, so

$$\lim_{r\to\infty} \sqrt[r]{\rho} = \lim_{r\to\infty} \sqrt[r]{\rho_1} = \frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}. \qquad \square$$

Table 4.1. Normalized convergence rates of polynomial filter estimators. p is the number of states on each agent, and r is the number of communication hops required to implement the estimator.

	Non-robust to ICs		Robust to ICs	
	$r = 1$	$r \to \infty$	$r = 2$	$r \to \infty$
$p = 1$	$\frac{1-\lambda_r}{1+\lambda_r}$	$\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}$	$\sqrt{\frac{1-\lambda_r^2}{1+\lambda_r^2}}$	$\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}$
$p = 2$	$\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}$	$\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}$	$\sqrt{\frac{1-\lambda_r}{1+\lambda_r}}$	$\frac{1-\sqrt{\lambda_r}}{1+\sqrt{\lambda_r}}$

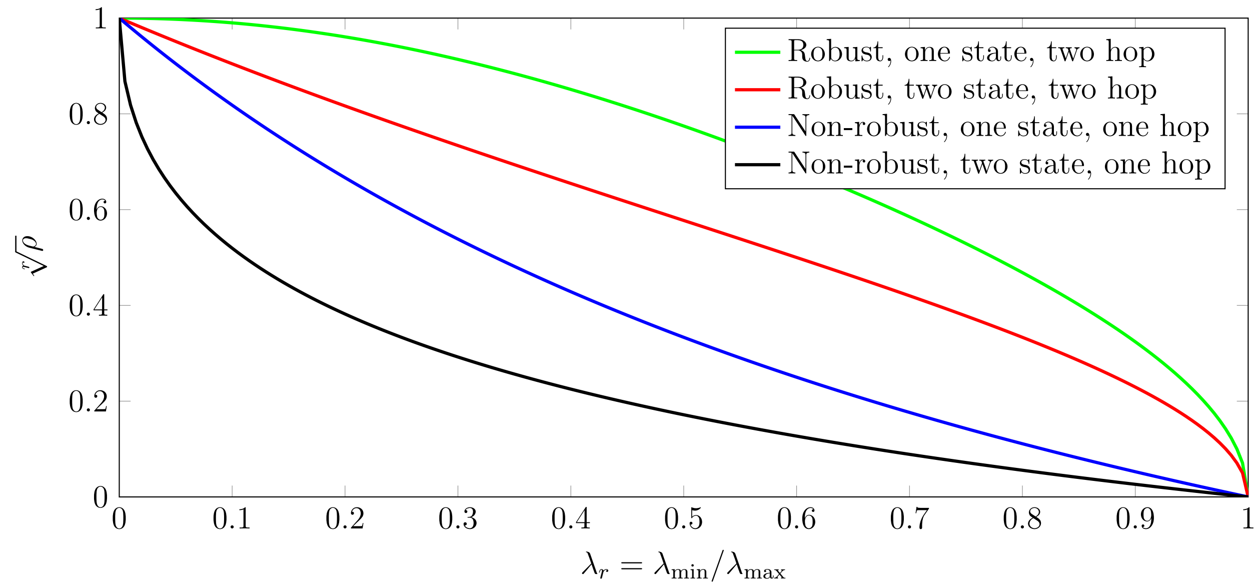


Figure 4.14. Normalized convergence rates of polynomial filter estimators. In the limit as the number of hops approaches infinity, the convergence rate of each polynomial filter approaches that of the non-robust, two-state, one-hop estimator.

The results for both non-robust and robust polynomial filter estimators with one and two states are summarized in Table 4.1 with the normalized convergence rates plotted in Fig. 4.14. Note that the non-robust estimators achieve better performance for small values of r, but their performance is the same in the limit as $r \to \infty$.

To summarize, we have shown the following:

- For estimators which are not robust to initial conditions, the optimal convergence rate can be obtained using only two internal states per agent and one-hop communication. (This is obtained by the P-2a estimator.)
- For estimators which are robust to initial conditions, the same convergence rate as the non-robust case is achieved in the limit as the number of communication hops approaches infinity (using either one or two internal states per agent).

4.5. Edge Estimator

In this section, we design an estimator that uses one-hop communication, is internally stable, is exact for constant inputs, is robust to initial conditions, and achieves the fastest known linear rate of convergence. Unlike the previous estimators where each agent has a fixed number of internal state variables, in this case each agent must have a state variable for each incoming edge. The estimator is unscalable to large dense graphs, although it can still be used when each agent has sufficient memory to store the variables for each edge (e.g., when the communication graph is sparse). Since the number of internal state variables in the system scales with m (the number of edges) instead of n (the number of agents), we call this the *edge estimator*.

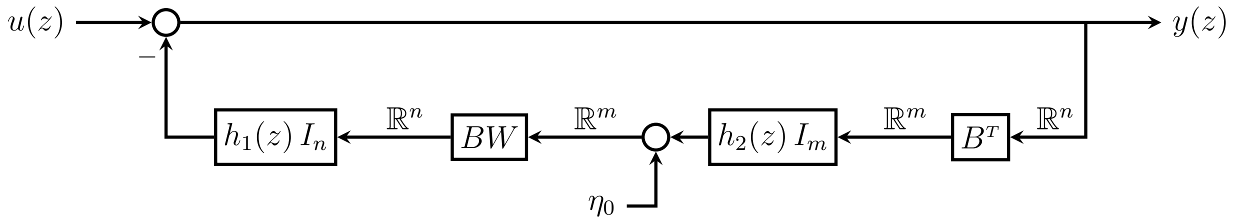


Figure 4.15. Block diagram of the edge estimator.

The block diagram of the edge estimator is shown in Fig. 4.15 with the space of the signals shown. The state equations for agent i are

$$\begin{aligned}
\nu^i_{k+1} &= a_1\,\nu^i_k + b_1 \sum_{j\in\mathcal{N}_{\text{in}}(i)} a_{ij}\left(c_2\,\eta^{i,j}_k + d_2\,(y^i_k - y^j_k)\right) \\
\eta^{i,j}_{k+1} &= a_2\,\eta^{i,j}_k + b_2\,(y^i_k - y^j_k), \quad j\in\mathcal{N}_{\text{in}}(i) \\
y^i_k &= u^i_k - \left(c_1\,\nu^i_k + d_1 \sum_{j\in\mathcal{N}_{\text{in}}(i)} a_{ij}\left(c_2\,\eta^{i,j}_k\right)\right)
\end{aligned} \tag{4.62}$$

where

$$h_1(z) = \left[\begin{array}{c|c} a_1 & b_1 \\ \hline c_1 & d_1 \end{array}\right] \qquad \text{and} \qquad h_2(z) = \left[\begin{array}{c|c} a_2 & b_2 \\ \hline c_2 & d_2 \end{array}\right]. \tag{4.63}$$

The number of internal state variables on agent i is $p_1 + p_2\,|\mathcal{N}_{\text{in}}(i)|$ where p_1 and p_2 are the degrees of $h_1(z)$ and $h_2(z)$, respectively. For the estimator to be well-posed, we require the product $h_1(z)\,h_2(z)$ to be strictly proper, i.e, $d_1\,d_2 = 0$.

From Chapter 2, we see that a Laplacian block is needed after the integrator (or more generally, the model of the input) for robustness to initial conditions and before the integrator for internal stability. However, we can get both properties using only one Laplacian matrix by factoring the Laplacian. When the graph is undirected, the Laplacian matrix can be factored using the oriented incidence matrix as $L = BWB^T$ where W is a diagonal matrix of edge weights. By moving the integrator block between B and B^T as shown in Fig. 4.15, we can get both internal stability and robustness to initial conditions. This can easily be seen by separating the system (see Section 2.6). Since $\mathbf{1}_n^T B = \mathbf{0}_m^T$, the separated system in the consensus direction is simply a gain of one; in particular, the output does not depend on the initial condition η_0. (The output also does not depend on ν_0 since $h_1(z)$ is chosen to be strictly stable.) Then the output converges to $\text{avg}(u)\mathbf{1}_n$. Since $B^T\mathbf{1}_n = 0_m$, the input to the integrator is zero in steady-state so the estimator is internally stable.

The edge estimator has the same transfer function as the proportional estimator in Fig. 4.4(a,b). Therefore, $h(z)$ can be chosen in the same way as the proportional estimator to optimize the convergence rate.

Edge-1. *The Edge-1 estimator is the estimator in Fig. 4.15 with*

$$h_1(z) = k_p \qquad \text{and} \qquad h_2(z) = \frac{1}{z-1}$$

where

$$k_p = \frac{2}{\lambda_{\max} + \lambda_{\min}} \qquad \text{and} \qquad \rho = \frac{\lambda_{\max} - \lambda_{\min}}{\lambda_{\max} + \lambda_{\min}}.$$

Edge-2. *The Edge-2 estimator is the estimator in Fig. 4.15 with*

$$h_1(z) = \frac{k_p}{z-\rho^2} \qquad \text{and} \qquad h_2(z) = \frac{z}{z-1}$$

where

$$k_p = \frac{4}{(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})^2} \qquad \text{and} \qquad \rho = \frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}}.$$

4.6. Nonlinear Estimator

All of the estimators described so far in this chapter have been linear. In this section, however, we propose a nonlinear estimator for dynamic average consensus. This estimator is exact for constant inputs, is time-invariant, is internally stable, uses one-hop local broadcast communication, and has the fastest known convergence rate (equivalent to the P-2 estimator, $\rho = (\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}})/(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})$). The estimator is also robust to initial conditions in a restricted sense; there is an open set of initial conditions from which the error converges linearly to zero. However, the set of initial conditions is not the entire space. In other words, the estimator is *locally* convergent, but not *globally* convergent. Note that we have previously used the term *robust to initial conditions* to denote both local and global convergence since they are equivalent for linear estimators. For the nonlinear estimator we must now distinguish the two types of robustness.

To develop the nonlinear estimator, first consider the proportional estimator in Fig. 4.4b. Recall that this estimator is globally convergent (since there is a Laplacian block after the integrator), but is not internally stable (since there is no Laplacian block before the integrator). The problem is that the integrator states grow unbounded when the outputs have converged to the average of the inputs. To fix this, we propose changing the state space on each agent from the plane to the cylinder, and in doing so introduce nonlinearities into the dynamics. To do this, we propose to have the integrator states take values on the compact manifold S^1 instead of the real line, where S^1 denotes the unit circle. We must modify the dynamics to make this work, but the result will be that the integrator states are automatically bounded regardless of the other signals in the system. In essence, we are proposing dynamic consensus via filtered oscillators, an extension of the unfiltered static oscillator consensus method introduced in [**38**, Ch. 2].

Let $\mathcal{I} : \mathbb{R} \to \mathbb{T}$ denote the covering projection $s \mapsto \mathcal{I}(s) = \exp\{s\sqrt{-1}\}$ for $s \in \mathbb{R}$. We identify S^1 with the unit circle $\mathbb{T}$ in the complex plane, but to achieve notational consistency we will write the group operator additively, using $\oplus$ and $\ominus$ to denote angle addition and subtraction. When used with vector arguments, we interpret $\mathcal{I}$, f, $\oplus$, and $\ominus$ as acting element-wise on each element of the vector to produce vector values.

We assume that $f \circ \mathcal{I}$ is odd, continuous, and satisfies $(f \circ \mathcal{I})'(0) = 1$. Other than these requirements, we are free to choose $f \circ \mathcal{I}$ as desired; a simple choice is $f \circ \mathcal{I} = \sin$.

Given a constant scaling parameter $\zeta > 0$, the nonlinear estimator has the form shown in Fig. 4.16 where

$$h_1(z) = \left[\begin{array}{c|c} a_1 & b_1 \\ \hline c_1 & d_1 \end{array}\right] \qquad \text{and} \qquad h_2(z) = \left[\begin{array}{c|c} a_2 & b_2 \\ \hline c_2 & d_2 \end{array}\right] \tag{4.64}$$

with $d_1\, d_2 = 0$ so that the system is well-posed. On agent i this takes the form

$$\begin{aligned} \nu_{k+1}^i &= a_1\, \nu_k^i + b_1 \sum_{j\in\mathcal{N}_{\text{in}}(i)} a_{ij}\, f(\psi_k^i - \psi_k^j) \\ \eta_{k+1}^i &= a_2\, \eta_k^i \oplus b_2\, \mathcal{I}\Big(\frac{y_k^i}{\zeta}\Big) \\ \psi_k^i &= c_2\, \eta_k^i \oplus d_2\, \mathcal{I}\Big(\frac{u_k^i}{\zeta} - c_1\, \nu_k^i\Big) \\ y_k^i &= u_k^i - \zeta\Big(c_1\, \nu_k^i + d_1 \sum_{j\in\mathcal{N}_{\text{in}}(i)} a_{ij}\, f(\psi_k^i - \psi_k^j)\Big) \end{aligned} \tag{4.65}$$

where η_k^i and ψ_k^i take values in $\mathbb{T}$, and ψ_k^i is transmitted to neighboring agents. Stacking the variables into vectors, we can write the system compactly as

$$\begin{aligned} \nu_{k+1} &= (a_1 \otimes I_n)\, \nu_k + (b_1 \otimes I_n)\, \mathcal{L}(\psi_k) \\ \eta_{k+1} &= (a_2 \otimes I_n)\, \eta_k \oplus (b_2 \otimes I_n)\, \mathcal{I}\Big(\frac{y_k}{\zeta}\Big) \\ \psi_k &= (c_2 \otimes I_n)\, \eta_k \oplus (d_2 \otimes I_n)\, \mathcal{I}\Big(\frac{u_k}{\zeta} - (c_1 \otimes I_n)\, \nu_k\Big) \\ y_k &= u_k - \zeta\big((c_1 \otimes I_n)\, \nu_k + (d_1 \otimes I_n)\, \mathcal{L}(\psi_k)\big) \end{aligned} \tag{4.66}$$

where $\psi = [\psi^1, \dots, \psi^n]^T \in \mathbb{T}^n$ (and similar for η).

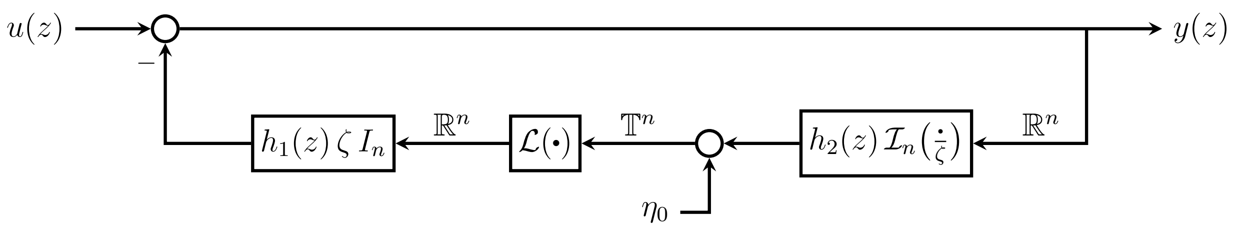


Figure 4.16. Block diagram of the nonlinear estimator.

Observe that the nonlinear estimator is similar to the proportional estimator in Fig. 4.4b—we have merely replaced the Laplacian and integrator blocks with nonlinear versions, introduced the scaling parameter ζ, and moved the Laplacian block between the two factors of $h(z) = h_1(z)\,h_2(z)$.

We need the scaling parameter ζ here because the nonlinear Laplacian operator $\mathcal{L}$ has a bounded image in $\mathbb{R}^n$ when Ω is compact. Indeed, the continuous function $|f|$ achieves a maximum value $f_{\max}$ on the compact set $\Omega = \mathbb{T}$, so the Laplacian is bounded as

$$\|\mathcal{L}(\cdot)\|_\infty \leq f_{\max}\|BW\|_\infty\,. \tag{4.67}$$

Using (4.66), we have that

$$\begin{aligned}(I_n - \Pi_n)\,u_k &= (y - \Pi_n u) + \zeta\big((c_1 \otimes I_n)\,\nu_k + (d_1 \otimes I_n)\,\mathcal{L}(\psi_k)\big)\\ &= e_k + \zeta\big((c_1 \otimes I_n)\,\nu_k + (d_1 \otimes I_n)\,\mathcal{L}(\psi_k)\big).\end{aligned} \tag{4.68}$$

The last term is the output of the filter $h_1(z)$ and therefore has ∞-norm bounded by the ℓ_1-norm of $h_1(z)$, denoted $\|h_1(z)\|_1$, multiplied by the ∞-norm of the input to $h_1(z)$. This gives the bound

$$\|(I_n - \Pi_n)\,u\|_\infty \leq \|e\|_\infty + \zeta\,\|h_1(z)\|_1\,\|\mathcal{L}(\cdot)\|_\infty \tag{4.69}$$

which implies

$$\zeta \geq \frac{\|(I_n - \Pi_n)\, u\|_\infty - \|e\|_\infty}{\|h_1(z)\|_1 \, f_{\max} \, \|BW\|_\infty}. \tag{4.70}$$

We can view this inequality as a necessary condition on the scaling parameter ζ. However, choosing ζ to satisfy (4.70) for a desired bound on the steady-state error requires knowledge of an upper bound on the differences between the inputs u^i and their global average $\mathbf{1}_n^T u$. Hence this estimator cannot achieve bounded error for general ramp inputs or other types of unbounded input signals, even when the estimator is cascaded, because eventually the bounded nonlinear Laplacian will be unable to produce large enough outputs. This is a clear disadvantage of the nonlinear estimator. However, this is not an issue so long as the input signals are bounded, as is the case in many practical applications.

Similar to the proportional estimator with two-states, we choose $h_1(z)$ and $h_2(z)$ so that the zeros of $1+\lambda\, h_1(z)\, h_2(z)$ are on $\rho\mathbb{T}$ for all $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. We also choose h_1 to be strictly proper so that the output only depends on the internal states (and not the Laplacian), and we choose h_2 to have a pole at $z = 1$ since this transfer function is implemented on the compact manifold S^1 (and therefore cannot become unbounded). This produces the NL-2 estimator summarized below.

NL-2. *The NL-2 (nonlinear, two-state) estimator is the estimator in Fig. 4.16 with*

$$h_1(z) = \frac{k_p}{z - \rho^2} \qquad \textit{and} \qquad h_2(z) = \frac{z}{z \ominus 1}$$

where

$$k_p = \frac{4}{(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})^2} \qquad \textit{and} \qquad \rho = \frac{\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}}}{\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}}}.$$

To analyze the NL-2 estimator, we assume for simplicity that there exists $b \in (0, \pi)$ such that $(f \circ \mathcal{I})(s) = s$ when $|s| \leq b$. In other words, we assume that the restriction of the 2π-periodic function $f \circ \mathcal{I}$ to the interval $[-b, b]$ is the identity map. This means that the NL-2 estimator will generate the same output as the linear P-2 estimator in Fig. 4.4 when all initial states are zero and the scaling parameter ζ is sufficiently large relative to the size of the inputs. In this case, we can show that if u is constant and ζ satisfies

$$\left\| u - u_{\text{ave}} \mathbf{1} \right\|_\infty \leq \frac{\zeta b \sqrt{\lambda_r}}{2\sqrt{n}} \tag{4.71}$$

where $\lambda_r = \lambda_{\min}/\lambda_{\max}$, then there is an open set of initial node states from which the estimator outputs y_k^i all converge to the global average $\mathbf{1}_n^T u$ as $k \to \infty$. Furthermore, the convergence is exponential with rate ρ (thus it is as fast as the P-2 estimator). Finally, simulations suggest that this open set of initial agent states (namely, the estimator's region of attraction) is large for sufficiently large values of ζ.

The results for the NL-2 estimator are summarized in the following theorem.

Theorem 16 (NL-2)**.** *Consider the NL-2 estimator. Let $\mathcal{G}$ be a constant, connected, undirected graph with Laplacian operator $\mathcal{L} : \mathbb{T}^n \to \mathbb{R}^n$ defined by $\mathcal{L}(x) = BWf(B^T x)$ where f is continuous. Suppose there exists $b \in (0, \pi)$ such that $(f \circ \mathcal{I})(s) = s$ when $|s| \leq b$ and that ζ is sufficiently large. Then there is an open set of initial states from which the output of the estimator converges to the average of the inputs. Furthermore, the rate of convergence is $\rho = (\sqrt{\lambda_{\max}} - \sqrt{\lambda_{\min}})/(\sqrt{\lambda_{\max}} + \sqrt{\lambda_{\min}})$.*

4.7. Summary

Many feedback estimators for dynamic average consensus have been presented in this chapter. To summarize the results, we give the properties of each estimator in Table 4.2 and plot the corresponding convergence rates in Fig. 4.17. Only the PI estimator has all of the desired properties in Problem 3. The convergence rate of the PI-4 is significantly faster than the PI-2, although not quite as fast as the fastest known methods (i.e., the convergence rate $\rho = (1-\sqrt{\lambda_r})/(1+\sqrt{\lambda_r})$). Furthermore, the PI-4 only requires four internal state variables on each agent, and each agent only broadcasts two scalar variables to its local neighbors at each iteration.

Table 4.2. Sumary of the properties of feedback estimators.

	Estimator	**Internal states on agent i**	**Transmission variables**	**Communication hops**	**Time-invariant**	**Scalable**	**Exact**	**Internally stable**	**Locally convergent**	**Globally convergent**
P	P-1a	1	1	1	✓	✓	✓	✓	✗	✗
	P-2a	2	1	1	✓	✓	✓	✓	✗	✗
	P-1b	1	1	1	✓	✓	✓	✗	✓	✓
	P-2b	2	1	1	✓	✓	✓	✗	✓	✓
	P-1c	1	1	1	✓	✓	✗	✗	✗	✗
	P-2c	2	1	1	✓	✓	✗	✗	✗	✗
PI	PI-2	2	2	1	✓	✓	✓	✓	✓	✓
	PI-4	4	2	1	✓	✓	✓	✓	✓	✓
PF	PF-NR-1-r	1	1	r	✓	✓	✓	✓	✗	✗
	PF-NR-2-r	2	1	r	✓	✓	✓	✓	✗	✗
	PF-R-1-r	1	1	r	✓	✓	✓	✓	✓	✓
	PF-R-2-r	2	1	r	✓	✓	✓	✓	✓	✓
Edge	Edge-1	$\lvert\mathcal{N}_{\text{in}}(i)\rvert$	1	1	✓	✗	✓	✓	✓	✓
	Edge-2	$1+\lvert\mathcal{N}_{\text{in}}(i)\rvert$	1	1	✓	✗	✓	✓	✓	✓
NL	NL-2	2	1	1	✓	✓	✓	✓	✓	✗

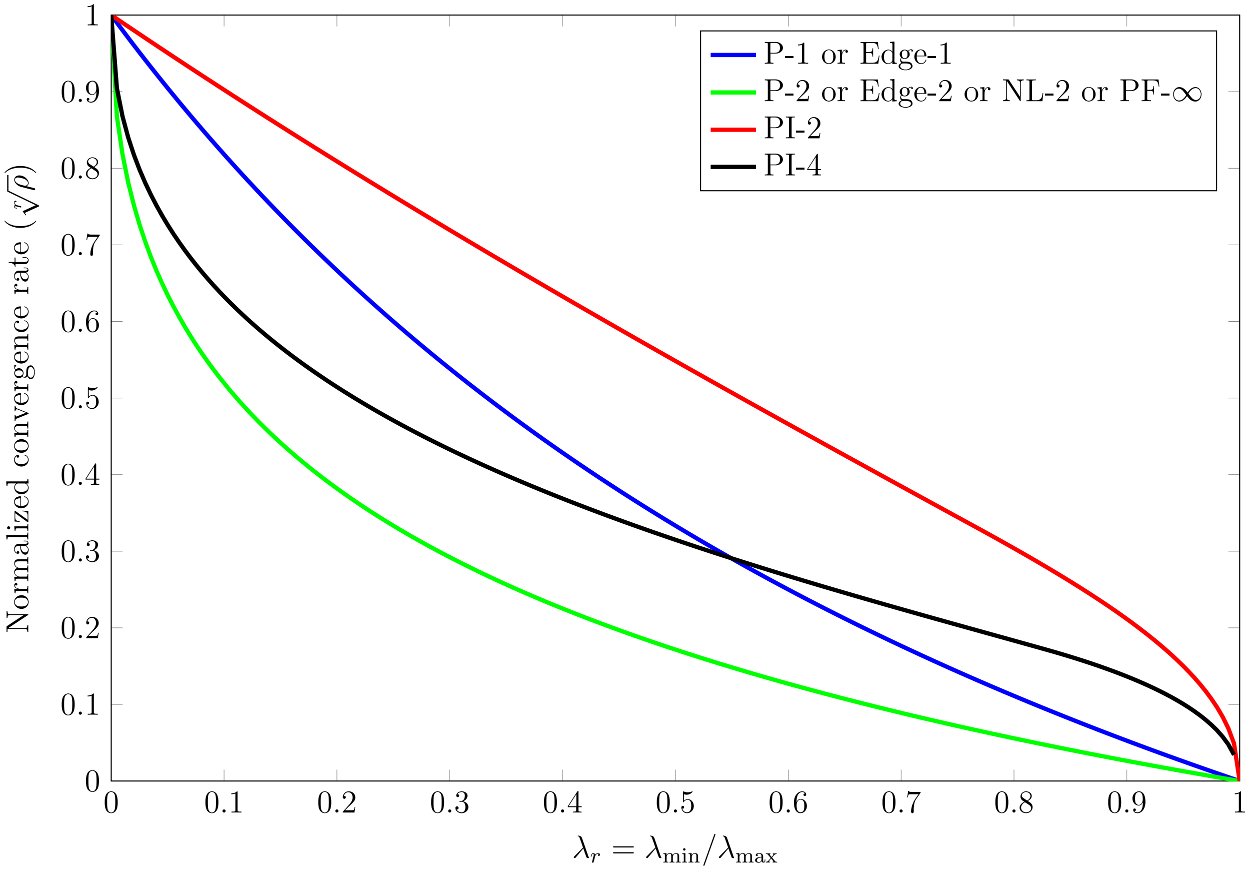


Figure 4.17. Summary of the convergence rates of feedback estimators. We use PF-∞ to denote any of the polynomial filter estimators (either robust or non-robust and one or two internal state variables) in the limit as $r \to \infty$.

CHAPTER 5

Estimators for Signals with Continuous Frequency Spectrum

In this chapter we design estimators to solve the dynamic average consensus problem when the frequency spectrum of the input signals is composed of a continuous band of frequencies. The design is given for frequency bands of the form $[0, \theta_c]$, but could be extended to include passband and highpass designs as well. We assume that the cutoff frequency is known; if unknown, the cutoff frequency may be estimated and the estimate used to design the algorithm. If the input signals contain multiple frequency bands, then multiple estimators, each designed for a single frequency band, may be cascaded together in series to track the average of the signals.

Unlike the estimators for signals with discrete frequency spectrum, the estimators designed in this chapter use a feedforward design. First, we show that the feedback estimators considered previously are unsuitable for tracking the average of bandlimited signals. We then propose a feedforward estimator which can achieve arbitrarily small steady-state error even when the communication graph is time-varying. We bound the steady-state error of the feedforward estimator for a general time-varying graph, and show that the error can be made arbitrarily small when the graph is connected, balanced, and has Laplacian matrix $L_k \in \mathbb{L}_n(\alpha)$ for some $\alpha < 1$ at each iteration. If the graph is constant and the graph topology is known, the steady-state error can be minimized by choosing the edge weights using the convex optimization problem specified in Problem 2 to minimize α. Furthermore, we show through simulations that the performance degrades gracefully as these assumptions are violated.

Therefore, we solve the following problem in this chapter.

Problem 5. *Suppose $\mathcal{G}$ is a time-varying graph which on average is connected, balanced, and has Laplacian matrix $L \in \mathbb{L}_n(\alpha)$ where $\alpha < 1$. Also, suppose the input signals are bandlimited with known cutoff frequency θ_c, i.e., $|u(e^{j\theta})| = 0$ for all $\theta \in [\theta_c, \pi]$. Design an estimator which*

(1) uses one-hop local broadcast communication,

(2) is scalable,

(3) is internally stable,

(4) is time-invariant,

(5) is robust to initial conditions,

(6) is robust to changes in the graph,

(7) and has bounded steady-state error.

Furthermore, the estimator should have the minimum attainable bound on the steady-state error.

In contrast to the problem in the previous chapter, here we prioritize minimizing the steady-state error instead of the convergence rate (since the steady-state error cannot be made zero in this case). There is a trade-off, however, between steady-state error and convergence rate as we will see. In particular, the settling time of the feedforward estimator increases as we tighten the bound on the steady-state error.

This chapter is organized as follows. First, we analyze cascaded versions of two proportional feedback estimators from Chapter 4 when the input signals are bandlimited, and it is shown that neither estimator is capable of solving Problem 5. Then we analyze several feedforward designs and show that there is a feedforward estimator which solves Problem 5 with arbitrarily small steady-state error.

5.1. Analysis

For this section, we relax the assumption of the communication graph being time-varying. When the communication graph is constant, the estimators we consider are linear time-invariant and can therefore be studied using standard frequency-domain techniques in control. In this case, the system is completely characterized by the transfer function from the input to the output, denoted $H(z, L)$. In this section, we give the block diagram of the general estimator structure considered and setup the tools needed to characterize the error of the estimator.

Denote the transfer function from the input to the error as

$$H_{\text{err}}(z, L) = H(z, L) - \Pi_n \tag{5.1}$$

where $\Pi_n := \mathbf{1}_n \mathbf{1}_n^T / n$ so that $e(z) = H_{\text{err}}(z, L)\, u(z)$. For undirected graphs, L is symmetric so the singular values of the error transfer function are

$$\sigma(\theta, \lambda) = \begin{cases} |H(e^{j\theta}, 0) - 1|, & \lambda = 0 \\ |H(e^{j\theta}, \lambda)|, & \lambda \in [\lambda_{\min}, \lambda_{\max}] \end{cases} \tag{5.2}$$

where the nonzero eigenvalues of L are in $[\lambda_{\min}, \lambda_{\max}]$. We analyze the response of the system in the passband ($\theta \in [0, \theta_c]$), stopband ($\theta \in (\theta_c, \pi]$), consensus direction ($\lambda = 0$), and disagreement directions ($\lambda \in [\lambda_{\min}, \lambda_{\max}]$) separately. The maximum singular value in each case is given in Table 5.1.

Table 5.1. Maximum singular values of the error transfer function in the consensus and disagreement directions and the passband and stopband.

	Consensus	Disagreement
Passband	$\sigma_{\text{cons}}^{\text{pass}} = \sup_{\theta \in [0,\theta_c]} \sigma(\theta, 0)$	$\sigma_{\text{dis}}^{\text{pass}} = \sup_{\substack{\lambda \in [\lambda_{\min},\lambda_{\max}] \\ \theta \in [0,\theta_c]}} \sigma(\theta, \lambda)$
Stopband	$\sigma_{\text{cons}}^{\text{stop}} = \sup_{\theta \in [\theta_c,\pi]} \sigma(\theta, 0)$	$\sigma_{\text{dis}}^{\text{stop}} = \sup_{\substack{\lambda \in [\lambda_{\min},\lambda_{\max}] \\ \theta \in [\theta_c,\pi]}} \sigma(\theta, \lambda)$

For a signal $x(z)$, the 2-norm in the passband and stopband are defined as

$$\|x(z)\|_2^{\text{pass}} = \sqrt{\frac{1}{\pi} \int_0^{\theta_c} x(e^{-j\theta})^T x(e^{j\theta}) \, d\theta} \tag{5.3}$$

$$\|x(z)\|_2^{\text{stop}} = \sqrt{\frac{1}{\pi} \int_{\theta_c}^{\pi} x(e^{-j\theta})^T x(e^{j\theta}) \, d\theta}. \tag{5.4}$$

For a system $H(z)$, we define ∞-norm in the passband and stopband as

$$\|H(z)\|_\infty^{\text{pass}} = \sup_{\theta \in [0,\theta_c]} \bar{\sigma}(H(e^{j\theta})) \tag{5.5}$$

$$\|H(z)\|_\infty^{\text{stop}} = \sup_{\theta \in [\theta_c,\pi]} \bar{\sigma}(H(e^{j\theta})) \tag{5.6}$$

where $\bar{\sigma}(H(e^{j\theta}))$ indicates the maximum singular value of $H(e^{j\theta})$. The maximum error can then be bounded using the maximum singular values and the size of the inputs as follows.

Lemma 8 (Error bound)**.** *Let $\mathcal{G}$ be a constant, connected, and undirected graph with Laplacian matrix L. Consider an estimator with transfer function $H(z, L)$ from the input $u(z)$ to the output $y(z)$. The maximum absolute steady-state error is bounded by*

$$\begin{aligned} \limsup_{k\to\infty} \ \|e_k\|_\infty \le \sigma_{cons}^{pass} \|\Pi \, u(z)\|_2^{pass} + \sigma_{cons}^{stop} \|\Pi \, u(z)\|_2^{stop} \\ + \sigma_{dis}^{pass} \|(I - \Pi) \, u(z)\|_2^{pass} + \sigma_{dis}^{stop} \|(I - \Pi) \, u(z)\|_2^{stop} \end{aligned} \tag{5.7}$$

where the singular values are defined in (5.2) *and Table 5.1.*

Proof. Since the ∞-norm is bounded by the 2-norm, $\|e_k\|_\infty \le \|e_k\|_2 = \sqrt{e_k^T e_k}$ which gives

$$\limsup_{k\to\infty} \|e_k\|_\infty^2 \le \limsup_{k\to\infty} e_k^T e_k \le \sum_{k=-\infty}^{\infty} e_k^T e_k.$$

Parseval's theorem states that

$$\sum_{k=-\infty}^{\infty} e_k^T e_k = \frac{1}{\pi}\int_0^\pi |e(e^{j\theta})|^2 d\theta = \|e(z)\|_2^2.$$

Combining these results gives $\limsup_{k\to\infty} \|e_k\|_\infty \le \|e(z)\|_2$. Decomposing the input into the consensus component $\Pi\, u$ and the disagreement component $(I - \Pi)\, u$,

$$\begin{aligned}\|e(z)\|_2 &= \|H_{\text{err}}(z, L)\,[\Pi + (I - \Pi)]\, u(z)\|_2 \\ &\le \|H_{\text{err}}(z, L)\,\Pi\, u(z)\|_2 + \|H_{\text{err}}(z, L)\,(I - \Pi)\, u(z)\|_2.\end{aligned}$$

Since $\|x(z)\|_2 \le \|x(z)\|_2^{\text{pass}} + \|x(z)\|_2^{\text{stop}}$ for any $x(z)$, each term can be separated into the passband and stopband,

$$\begin{aligned}\|e(z)\|_2 \le{}& \|H_{\text{err}}(z, L)\,\Pi\, u(z)\|_2^{\text{pass}} + \|H_{\text{err}}(z, L)\,\Pi\, u(z)\|_2^{\text{stop}} \\ &+ \|H_{\text{err}}(z, L)\,(I - \Pi)\, u(z)\|_2^{\text{pass}} + \|H_{\text{err}}(z, L)\,(I - \Pi)\, u(z)\|_2^{\text{stop}}.\end{aligned}$$

Since the graph is undirected, the singular values of $H_{\text{err}}(e^{j\theta}, L)$ are given by (5.2) with the maximum singular values in Table 5.1. This gives the bound in (5.7). □

Lemma 8 shows that we want to minimize the singular values of the error transfer function in order for the estimator to solve the dynamic average consensus problem with small steady-state error.

If the input signals are bandlimited with cutoff frequency θ_c, then $\|\Pi\, u\|_2^{\text{stop}} = 0 = \|(I - \Pi)\, u\|_2^{\text{stop}}$ in Lemma 8. When implemented using finite arithmetic, however, rounding errors create small high-frequency components in the signals. These high frequencies can cause the error to be large if the stopband singular values ($\sigma_{\text{cons}}^{\text{stop}}$ and $\sigma_{\text{dis}}^{\text{stop}}$) are too large.

Many dynamic average consensus estimators use a cascade structure [**23, 25, 27, 66**]. Throughout the rest of this paper, we consider the cascade block diagram structure in Fig. 5.1. The transfer function from the input to the output is

$$H(z, L) = h_{\text{pre}}(z)\, [G(z, L)]^{\ell} \tag{5.8}$$

where $h_{\text{pre}}(z)$ is a prefilter and ℓ is the number of stages. The prefilter is applied directly to the input signals and can be implemented independently on each agent without requiring any communication with neighbors. The main estimator $G(z, L)$ involves both internal processing and communication with local neighbors and is applied ℓ times in series. In order for the estimator to be implementable using one-hop local broadcast communication, there must be a strictly proper transfer function between each Laplacian block. This requires $G(z, L)$ to be a one-hop estimator with no direct feedthrough of the Laplacian. In other words, $G(z, L)$ must have the form

$$G(z, L) = \left[\begin{array}{c|c} a(L) & b(L) \\ \hline c(L) & d(L) \end{array}\right] = \left[\begin{array}{c|c} a_0 & b_0 \\ \hline c_0 & d_0 \end{array}\right] \otimes I_n + \left[\begin{array}{c|c} a_1 & b_1 \\ \hline c_1 & 0 \end{array}\right] \otimes L. \tag{5.9}$$

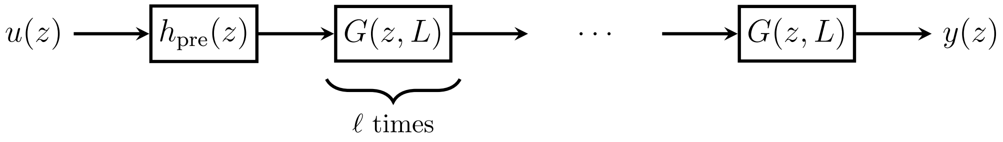


Figure 5.1. Block diagram of the general cascade estimator structure with transfer function from the input to the output $H(z, L) = h_{\text{pre}}(z)\,[G(z, L)]^{\ell}$.

We now present some estimator designs, i.e., specific choices for the filters $h_{\text{pre}}(z)$ and $G(z, L)$. The designs are separated into two categories based on whether $G(z, L)$ applies the Laplacian in the feedback or feedforward path.

5.2. Feedback Estimators

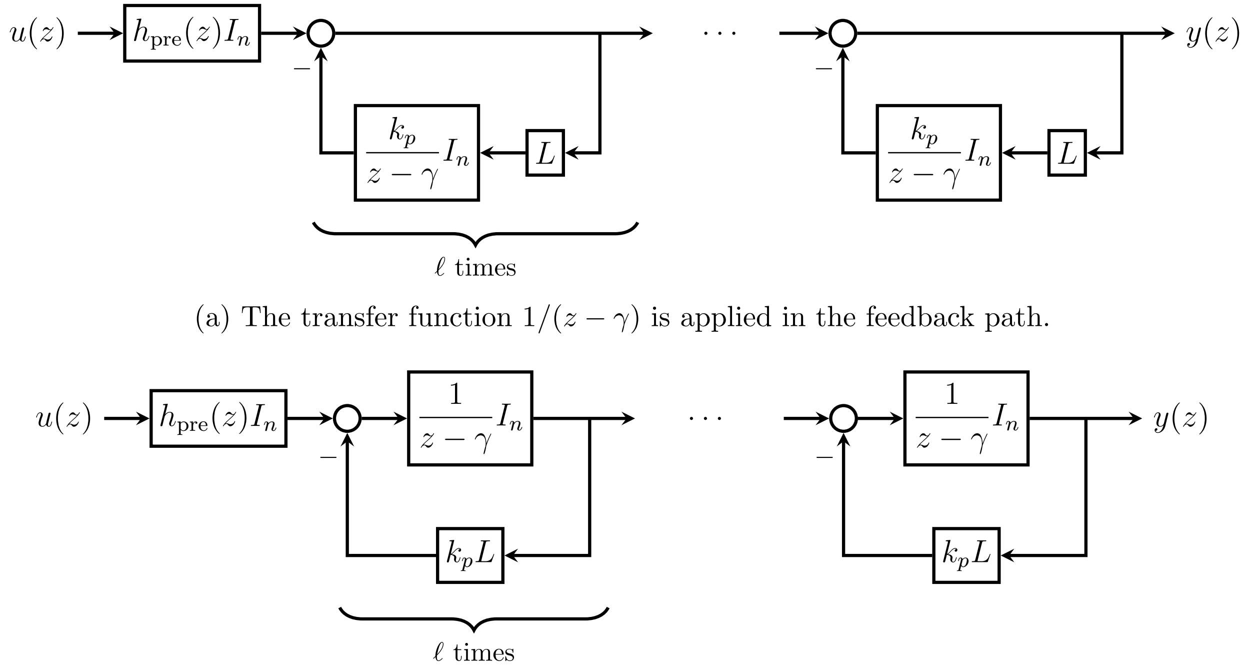
We consider two feedback estimators in this section and show that neither adequately solves the dynamic average consensus problem in Problem 5. The block diagrams of the feedback estimators considered are shown in Fig. 5.2. Both estimators use a cascade of simple one-variable feedback estimators from Chapter 4.

Both feedback estimators in Fig. 5.2 are generalizations of previously considered estimators. Specifically, we show in Table 5.2 how these estimators compare to those in the literature.

Table 5.2. Specific feedback estimators of the form considered in Fig. 5.2.

Estimator	**Fig.**	**Parameters**
Freeman, Yang, and Lynch [**19**]	5.2a	$h_{\text{pre}}(z) = 1$, $\ell = 1$
Kia, Cortés, and Martínez [**24**]	5.2a	$h_{\text{pre}}(z) = 1$, $\ell = 1$ (and pole/zero cancellation)
Zhu and Martínez [**23**]	5.2b	$h_{\text{pre}}(z) = (1 - z^{-1})^{\ell}$, $\gamma = 1$, $k_p = 1$
Franceschelli and Gasparri [**25**]	5.2b	$h_{\text{pre}}(z) = (1 - \gamma)^{\ell}$

We now give a detailed analysis of each of the feedback estimators in Fig. 5.2.



(a) The transfer function $1/(z - \gamma)$ is applied in the feedback path.

(b) The transfer function $1/(z - \gamma)$ is applied in the forward path.

Figure 5.2. Block diagrams of cascaded feedback estimators.

5.2.1. Analysis of the estimator in Fig. 5.2a

For the separated system of the estimator in Fig. 5.2a, the transfer function of each stage is

$$G(z, \lambda) = \frac{z - \gamma}{z - (\gamma - k_p \lambda)}, \tag{5.10}$$

and the transfer function from the input to the output is $H(z, \lambda) = h_{\text{pre}}(z)\,[G(z, \lambda)]^{\ell}$.

Lemma 9. *The convergence rate of the both estimators in Fig. 5.2 is*

$$\rho = \max\{\rho_{pre}, |\gamma|, |\gamma - k_p \lambda_{\max}|\} \tag{5.11}$$

where ρ_{pre} is the convergence rate of $h_{pre}(z)$.

Proof. Using the separated system, both estimators have ℓ closed-loop poles at $z = \gamma - k_p\lambda$ where $\lambda \in \{0\} \cup [\lambda_{\min}, \lambda_{\max}]$ along with the poles of $h_{\text{pre}}(z)$. The convergence rate is the maximum modulus of the closed-loop poles, which gives the result. □

Theorem 17. *Consider the estimator in Fig. 5.2a with $k_p > 0$. The maximum singular values of the error transfer function in Table 5.1 are*

$$\sigma_{cons}^{pass} = \|1 - h_{pre}(z)\|_\infty^{pass} \tag{5.12}$$

$$\sigma_{cons}^{stop} = \|1 - h_{pre}(z)\|_\infty^{stop} \tag{5.13}$$

$$\sigma_{dis}^{pass} = \|h_{pre}(z)\|_\infty^{pass} \left(\frac{1 - 2\gamma\cos\theta_c + \gamma^2}{1 - 2(\gamma - k_p\lambda^*)\cos\theta_c + (\gamma - k_p\lambda^*)^2}\right)^{\ell/2} \tag{5.14}$$

$$\sigma_{dis}^{stop} = \|h_{pre}(z)\|_\infty^{stop} \left(\frac{1 + \gamma}{1 + \gamma - k_p\lambda_{\max}}\right)^{\ell} \tag{5.15}$$

where

$$\lambda^* = \begin{cases} \lambda_{\min}, & (\gamma - \cos\theta_c)/k_p \le \lambda_{\min}, \\ (\gamma - \cos\theta_c)/k_p, & \lambda_{\min} < (\gamma - \cos\theta_c)/k_p < \lambda_{\max} \\ \lambda_{\max}, & \lambda_{\max} \le (\gamma - \cos\theta_c)/k_p. \end{cases} \tag{5.16}$$

Proof. The transfer function of the consensus system is $H(z, 0) = h_{\text{pre}}(z)$ which gives the singular values in the consensus direction. The singular values of the error transfer function in (5.2) are

$$\sigma(\theta, \lambda) = \left(\frac{1 - 2\gamma\cos\theta + \gamma^2}{1 - 2(\gamma - k_p\lambda)\cos\theta + (\gamma - k_p\lambda)^2}\right)^{\ell/2} \tag{5.17}$$

for $\lambda \in [\lambda_{\min}, \lambda_{\max}]$. In the passband, the maximum must occur either at an endpoint or when the derivative is zero. Solving $0 = \frac{\partial}{\partial\theta}\sigma$ gives $\theta = 0$, but $\frac{\partial^2}{\partial\theta^2}\sigma|_{\theta=0} > 0$, so this is a local minimum. The maximum must then occur at the endpoint $\theta = \theta_c$. Similarly, we maximize over λ by solving $0 = \frac{\partial}{\partial\lambda}\sigma$ which gives $\lambda = (\gamma - \cos\theta)/k_p$. This is a local maximum since $\frac{\partial^2}{\partial\theta^2}\sigma|_{\lambda=(\gamma-\cos\theta)/k_p} < 0$. The maximum occurs at $\lambda = (\gamma - \cos\theta)/k_p$ if $(\gamma - \cos\theta)/k_p \in [\lambda_{\min}, \lambda_{\max}]$, and otherwise occurs at the endpoint closest to $(\gamma - \cos\theta)/k_p$.

In the stopband, the maximum singular value occurs when $\theta = \pi$ and $\lambda = \lambda_{\max}$. □

Corollary 4. *For the estimator in Fig. 5.2a with* $k_p = 2\gamma/(\lambda_{\min} + \lambda_{\max})$ *and* $h_{pre}(z) = 1$, *the convergence rate is* γ, *and* $\sigma_{dis}^{pass} < 1$ *if*

$$0 < \gamma < \left(1 + \frac{\lambda_{\min}}{\lambda_{\max}}\right) \cos\theta_c. \tag{5.18}$$

Proof. The convergence rate can be found using Lemma 9. If $\lambda^* = (\gamma - \cos\theta_c)/k_p$ or $\lambda_{\max}$ in Theorem 17, then $\sigma_{\text{dis}}^{\text{pass}} \geq 1$. Setting $\sigma_{\text{dis}}^{\text{pass}} < 1$ in (5.14) with $\lambda^* = \lambda_{\min}$ gives the result. □

The choice $k_p = 2\gamma/(\lambda_{\min} + \lambda_{\max})$ in Corollary 4 minimizes the convergence rate in the disagreement directions. Using this choice, however, restricts the set of graphs and input signals that the estimator is capable of tracking with small error since (5.18) must be satisfied. In particular, the estimator using this choice for k_p cannot track bandlimited signals with cutoff frequency $\theta_c \geq \pi/2$. For other parameter choices, however, the steady-state error can be made arbitrarily small as shown in the following corollary.

Corollary 5. *Consider the estimator in Fig. 5.2a with* $h_{pre}(z) = 1$, *and let* $\mathcal{G}$ *be a connected, undirected graph with nonzero Laplacian eigenvalues in* $[\lambda_{\min}, \lambda_{\max}]$. *Then the* *steady-state error can be made arbitrarily small if* $\cos\theta_c > \gamma - k_p\lambda_{\min}/2$*,* ℓ *is made large enough, and exact arithmetic is used.*

Proof. Since $h_{\text{pre}}(z) = 1$, we have $\sigma_{\text{cons}}^{\text{pass}} = \sigma_{\text{cons}}^{\text{stop}} = 0$. The error in the disagreement directions in the stopband can be made arbitrarily small since $\cos\theta_c > \gamma - k_p\lambda_{\min}/2$, i.e.,

$$\lim_{\ell\to\infty} \sigma_{\text{dis}}^{\text{pass}} = 0. \tag{5.19}$$

Using exact arithmetic, we have $\|u(z)\|_2^{\text{stop}} = 0$. Then applying Lemma 8, the steady-state error can be made arbitrarily small. □

Lemma 9 along with Corollaries 4 and 5 highlight the trade-off between convergence rate and steady-state error. The convergence rate is optimized in Corollary 4, but the set of graphs and input signals that the estimator can track with small error is limited by (5.18). The parameter choices in Corollary 5 allow the estimator to have arbitrarily small steady-state error, but the convergence rate becomes slow for graphs with low connectivity ($\lambda_{\min}/\lambda_{\max}$ small) and for fast time-varying input signals (θ_c large).

Simulations of the estimator in Fig. 5.2a are shown in Fig. 5.3. The error converges close to zero when the graph is constant. The graph changes at iteration 500, and the error converges back towards zero after a transient since the estimator is robust to initial conditions. After iteration 800, however, the graph changes at every iteration; the transient does not have time to decay and the error remains large. This indicates that the estimator is not robust to changes in the graph.

The estimator is not robust to changes in the graph because the steady-state values of the internal states depend on the Laplacian matrix. When the graph changes, the internal states must converge to different values causing a transient.

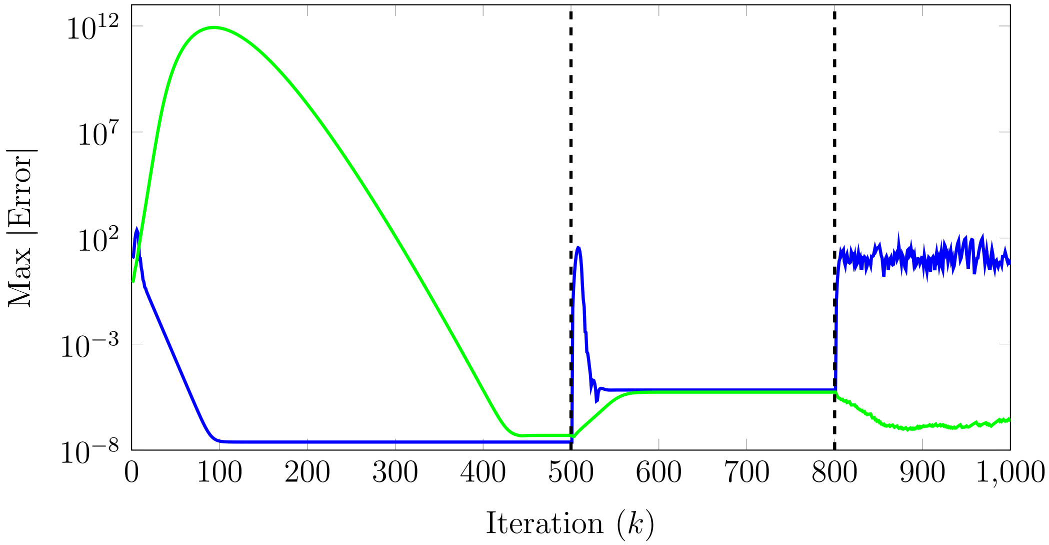


Figure 5.3. Simulation of the feedback estimators in Fig. 5.2a (blue) using $h_{\text{pre}}(z) = 1$ and Fig. 5.2b (green) using $h_{\text{pre}}(z) = (1-\gamma)^{\ell}$ with parameters $\theta_c = 0$, $\ell = 20$, and $\gamma = k_p = 0.8$. The graph changes at iteration 500 and then every iteration past 800.

To summarize the results for the estimator in Fig. 5.2a, the steady-state error can be made arbitrarily small by making ℓ large, and there is a trade-off between steady-state error and convergence rate that can be exploited through different choices of the parameters γ and k_p. However, the estimator is not robust to changes in the graph.

5.2.2. Analysis of the estimator in Fig. 5.2b

For the separated system of the estimator in Fig. 5.2b, the transfer function of each stage is

$$G(z,\lambda) = \frac{1}{z - (\gamma - k_p \lambda)}, \tag{5.20}$$

and the transfer function from the input to the output is $H(z,\lambda) = h_{\text{pre}}(z)\,[G(z,\lambda)]^{\ell}$.

Theorem 18. *Consider the estimator in Fig. 5.2b with* $k_p > 0$. *The maximum singular values of the error transfer function in Table 5.1 are*

$$\sigma_{cons}^{pass} = \left\| 1 - h_{pre}(z) \left(\frac{1}{z-\gamma} \right)^{\ell} \right\|_{\infty}^{pass} \tag{5.21}$$

$$\sigma_{cons}^{stop} = \left\| 1 - h_{pre}(z) \left(\frac{1}{z-\gamma} \right)^{\ell} \right\|_{\infty}^{stop} \tag{5.22}$$

$$\sigma_{dis}^{pass} = \| h_{pre}(z) \|_{\infty}^{pass} \max\{\sigma(0, \lambda_{\min}), \sigma(\theta_c, \lambda^*)\} \tag{5.23}$$

$$\sigma_{dis}^{stop} = \| h_{pre}(z) \|_{\infty}^{stop} \left(\frac{1}{1 + \gamma - k_p \lambda_{\max}} \right)^{\ell} \tag{5.24}$$

where

$$\sigma(\theta, \lambda) = \left(\frac{1}{1 - 2(\gamma - k_p\lambda)\cos\theta + (\gamma - k_p\lambda)^2} \right)^{\ell/2} \tag{5.25}$$

and λ^* *is given by* (5.16).

Proof. Equations (5.21) and (5.22) follow straight from the definition of the maximum singular values. In the disagreement directions, the singular values of $[G(z, \lambda)]^{\ell}$ are given by (5.25). The saddle points in the passband which are potential optima occur at $(\theta, \lambda) = (0, \lambda_{\min})$ and (θ_c, λ^*). In the stopband, the maximum occurs at $(\theta, \lambda) = (\pi, \lambda_{\max})$. □

The prefilter can be used to make the maximum singular value in the consensus direction small. Ideally, we would like to use the prefilter $h_{\text{pre}}(z) = (z - \gamma)^{\ell}$ so that the maximum singular value in the consensus direction is zero. The ideal prefilter is non-causal, but it could be approximated in the passband by a causal filter.

Corollary 6. *Consider the estimator in Fig. 5.2b, and let* $\mathcal{G}$ *be a connected, undirected graph with nonzero Laplacian eigenvalues in* $[\lambda_{\min}, \lambda_{\max}]$. *Then the steady-state error can be* *made arbitrarily small if the prefilter is designed such that* σ_{cons}^{pass} *in* (5.21) *is arbitrarily small,* $\|h_{pre}(z)\|_\infty$ *is bounded,* $\cos\theta_c > \gamma - k_p\lambda_{\min}/2$*, and* ℓ *is made large enough.*

Proof. Designing the prefilter such that $\sigma_{\text{cons}}^{\text{pass}}$ is arbitrarily small, we have $h_{\text{pre}}(z) \approx (z-\gamma)^\ell$ in the passband. The system transfer function in the passband is then

$$H(z,\lambda) \approx \left(\frac{z-\gamma}{z-(\gamma-k_p\lambda)}\right)^\ell$$

which is the same as the estimator in Fig. 5.2a. Therefore, the singular values are also the same so we can apply the results from Corollary 5 which gives the result. □

Corollary 6 shows that the steady-state error can be made arbitrarily small by designing $h_{\text{pre}}(z)$ to approximate the ideal prefilter $(z-\gamma)^\ell$ in the passband. Although the ideal prefilter is non-causal, it can be approximated arbitrarily closely in the passband by a causal filter.

Simulations of the estimator in Fig. 5.2b are shown in Fig. 5.3. The error converges close to zero and remains small, even after the graph changes. Since the same results hold for many simulations with various values of γ and k_p so long as $0 < k_p \leq \gamma/\lambda_{\max}$, we make the following conjecture.

Conjecture 1. *The estimator in Fig. 5.2b is robust to changes in the graph if* $0 < k_p \leq \gamma/\lambda_{\max}$.

To summarize the results for the estimator in Fig. 5.2b, the steady-state error can be made arbitrarily small if the following hold:

- ℓ is large enough,
- the prefilter approximates $(z-\gamma)^\ell$ arbitrarily closely in the passband (and is finite in the stopband), and

- the parameters are chosen such that $\cos\theta_c > \gamma - k_p\lambda_{\min}/2$.

Furthermore, the estimator is robust to changes in the graph if $0 < k_p \leq \gamma/\lambda_{\max}$.

In order for the estimator to achieve small steady-state error and be robust to changes in the graph, we need $\cos\theta_c > \gamma - k_p\lambda_{\min}/2$ and $0 < k_p \leq \gamma/\lambda_{\max}$. These conditions imply that $\theta_c < \pi/2$ which limits the cutoff frequency that can be used. Therefore, the estimator is not capable of tracking bandlimited signals with cutoff frequency $\theta_c < \pi$ with arbitrarily small error over time-varying graphs.

The estimator in Fig. 5.2a can have arbitrarily small steady-state error for arbitrarily fast time-varying input signals, but is not robust to changes in the graph. The estimator in Fig. 5.2b can be robust to changes in the graph and have arbitrarily small steady-state error, but can only achieve both when $\theta_c < \pi/2$. A feedforward estimator is proposed in the following section which is robust to changes in the graph and has arbitrarily small steady-state error for bandlimited signals with cutoff frequency $\theta_c < \pi$.

5.3. Feedforward Estimators

In this section, an estimator for dynamic average tracking is designed where $G(z, L)$ applies the Laplacian in the forward path. The estimator is:

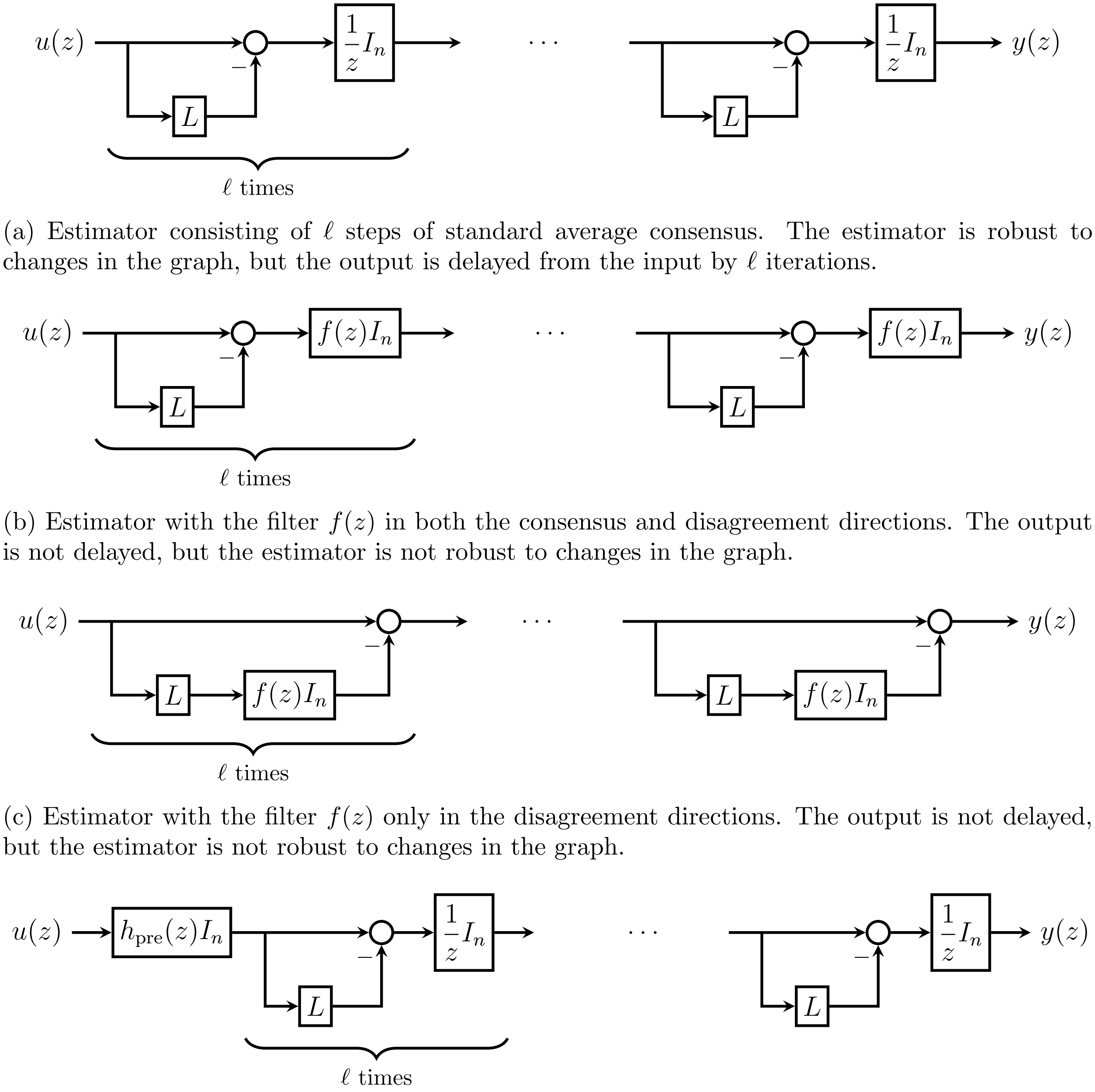
(1) robust to initial conditions,

(2) robust to changes in balanced graphs,

(3) capable of tracking bandlimited signals with cutoff frequency $\theta_c < \pi$ with arbitrarily small steady-state error.

None of the feedback estimators described in the previous section have all of these properties.

To design the estimator, first consider the estimator shown in Fig. 5.4a which simply multiplies the input by the matrix $(I - L)^\ell$ where $I - L$ is referred to as the consensus



(a) Estimator consisting of ℓ steps of standard average consensus. The estimator is robust to changes in the graph, but the output is delayed from the input by ℓ iterations.

(b) Estimator with the filter $f(z)$ in both the consensus and disagreement directions. The output is not delayed, but the estimator is not robust to changes in the graph.

(c) Estimator with the filter $f(z)$ only in the disagreement directions. The output is not delayed, but the estimator is not robust to changes in the graph.

(d) Estimator where the prefilter $h_{\text{pre}}(z) = [z\,f(z)]^{\ell}$ is implemented *before* passing through the graph Laplacian. The output is not delayed and the estimator is robust to changes in the graph.

Figure 5.4. Block diagrams of the cascaded feedforward estimators.

matrix. Each iteration requires time since multiplication by L requires communication among neighbors, so there is a delay of $1/z$ between each iteration. This design works well for *any* inputs (not just bandlimited inputs), is robust to initial conditions, is robust to changes in the graph, and can be applied to balanced graphs. The drawback, however, is that the output is delayed by ℓ steps from the input, so achieving a better estimate requires more delay.

To fix the delay, we could replace $1/z$ with a filter $f(z)$ as shown in Fig. 5.4b. The frequency response of the filter $f(z)$ should be designed to approximate unity for inputs with frequencies in $[0, \theta_c]$ where θ_c is the cutoff frequency of the input signals. Then the transfer function approximates $(I - L)^\ell$ in the passband. We could choose $f(z) = 1$, but this results in an ℓ-hop estimator. In this case, each iteration would require ℓ rounds of communication to be done sequentially since the result of each round is needed for the next. This is due to Laplacian blocks being directly connected in the block diagram without any delay between them. To prevent this, we require $f(z)$ to be strictly proper so that there is no direct feedthrough between Laplacian blocks. The estimator can then be implemented in one hop, meaning that each agent can broadcast all of its information in a single packet at each iteration. This fixes the delay problem, but the estimator is not robust to changes in the graph.

The design in Fig. 5.4b filters both the consensus and disagreement directions. Since we only require a strictly proper filter between consecutive Laplacian blocks, we could also place the filter directly after each Laplacian block as in Fig. 5.4c. A benefit of this design is that it has zero error for any signals (not only bandlimited signals) which are common to all agents since the output is directly connected to the input in the consensus direction. This estimator also has no delay at the output, but is still not robust to changes in the graph.

To see why the estimators in Figs. 5.4b and 5.4c are not robust to changes in the graph, consider implementing the estimators on a time-varying graph. In each stage, the input to

$f(z)$ is a linear function of the output of the previous stage. Linear functions of bandlimited signals are also bandlimited. When the graph changes, however, different linear combinations are taken at each iteration. This produces signals which are not bandlimited, in which case $f(z)$ does not approximate unity and the error is large.

We can make an estimator with the same transfer function as that in Fig. 5.4b which *is* robust to changes in the graph by applying the filter $f(z)$ before the signal passes through the Laplacian as shown in Fig. 5.4d. After the prefilter $[z\,f(z)]^\ell$, the rest of the estimator is identical to that in Fig. 5.4a which works for any inputs (not just bandlimited inputs) and is robust to changes in the graph. The problem with the estimator in Fig. 5.4a was that the output was delayed, but this is offset in the estimator in Fig. 5.4d by the prefilter. Therefore, we propose the estimator in Fig. 5.4d to solve the dynamic average consensus problem without delay and over time-varying graphs.

The prefilter can be constructed by designing the filter $f(z)$ to be strictly proper and to approximate unity for $z = \exp\{j\theta\}$, $\theta \in [0, \theta_c]$, and then setting $h_{\text{pre}}(z) = [z\,f(z)]^\ell$.

In many applications, power is a valuable resource for each agent. Communication with neighbors can consume the majority of the available power, and should therefore only be used when necessary. Since the prefilter occurs before any communication, each agent can run its own local prefilter without communicating during the initial transient period to save power. Once the prefilter has converged, the agent begins transmitting information to local neighbors and the estimator only requires a finite number of iterations (specifically, ℓ iterations) to reach steady-state.

We now analyze the feedforward estimator in Fig. 5.4d.

Lemma 10. *The estimator in Fig. 5.4d with $h_{pre}(z)$ strictly stable is robust to initial conditions.*

Proof. The system has ℓ poles at $z = 0$ and shares the same poles as the prefilter $h_{\text{pre}}(z)$. Since the prefilter is strictly stable, all of the system poles are strictly inside the unit circle, so the steady-state value is independent of the initial conditions. □

Theorem 19 (Error bound of the feedforward estimator)**.** *Consider the estimator in Fig. 5.4d. Let $\mathcal{G}$ be any (possibly time-varying) graph. Then the maximum absolute steady-state error is bounded by*

$$\begin{aligned}\limsup_{k\to\infty} \|e_k\|_\infty \le \sigma_{cons}^{pass} \|\Pi\, u(z)\|_2^{pass} + \sigma_{cons}^{stop} \|\Pi\, u(z)\|_2^{stop} \\ + \delta \left(\sigma_{dis}^{pass} \|(I-\Pi)\, u(z)\|_2^{pass} + \sigma_{dis}^{stop} \|(I-\Pi)\, u(z)\|_2^{stop}\right)\end{aligned} \tag{5.26}$$

where

$$\begin{aligned}\sigma_{cons}^{pass} &= \|z^\ell - h_{pre}(z)\|_\infty^{pass} & \sigma_{dis}^{pass} &= \|h_{pre}(z)\|_\infty^{pass} \\ \sigma_{cons}^{stop} &= \|z^\ell - h_{pre}(z)\|_\infty^{stop} & \sigma_{dis}^{stop} &= \|h_{pre}(z)\|_\infty^{stop}\end{aligned} \tag{5.27}$$

and

$$\delta = \|\mathcal{W}_k^\ell - \Pi\|_2 \tag{5.28}$$

with

$$\mathcal{W}_k^\ell := (I - L_{k-1})(I - L_{k-2}) \cdots (I - L_{k-\ell}). \tag{5.29}$$

Proof. Denote the output of the prefilter due to the input u as $\tilde{u}(z) = [I_n \otimes h_{\text{pre}}(z)]\, u(z)$. Then the output of the estimator is $y_k = \mathcal{W}_k^\ell\, \tilde{u}_{k-\ell}$ and the error is $e_k = y_k - \Pi\, u_k$.

Since $L_k \mathbf{1}_n = \mathbf{0}_n$, we have $(\mathcal{W}_k^\ell - \Pi)\,\Pi = \mathbf{0}_n$. Using the fact that $\|x\|_\infty \leq \|x\|_2$ for all $x \in \mathbb{R}^n$, the error is bounded by

$$\begin{aligned}
\|e_k\|_\infty &= \|y_k - \Pi\, u_k\|_\infty \\
&= \|(\mathcal{W}_k^\ell - \Pi)(I - \Pi)\,\tilde{u}_{k-\ell} + \Pi\,(\tilde{u}_{k-\ell} - u_k)\|_\infty \\
&\leq \|\mathcal{W}_k^\ell - \Pi\|_2\, \|(I - \Pi)\,\tilde{u}_{k-\ell}\|_2 + \|\Pi\,(\tilde{u}_{k-\ell} - u_k)\|_2.
\end{aligned}$$

The transfer function from u_k to $\tilde{u}_{k-\ell}$ is $I_n \otimes \big(z^{-\ell}\, h_{\text{pre}}(z)\big)$, and the transfer function from u_k to $\tilde{u}_{k-\ell} - u_k$ is $(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z)) - I_n$. Since $\|x_k\|_2 \leq \|x(z)\|_2$ for any signal $x \in \ell_{2e}$, we have

$$\|e_k\|_\infty \leq \big\|\mathcal{W}_k^\ell - \Pi\|_2 \,\big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z)\big)\,(I - \Pi)\,u(z)\big\|_2 + \big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z) - I_n\big)\,\Pi\, u(z)\big\|_2.$$

Since $\|x(z)\|_2 \leq \|x(z)\|_2^{\text{pass}} + \|x(z)\|_2^{\text{stop}}$ for any $x(z)$, we can split the terms into the passband and stopband,

$$\begin{aligned}
\|e_k\|_\infty \leq \delta \Big(&\big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z)\big)\,(I - \Pi)\,u(z)\big\|_2^{\text{pass}} + \big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z)\big)\,(I - \Pi)\,u(z)\big\|_2^{\text{stop}}\Big) \\
&+ \big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z) - I_n\big)\,\Pi\, u(z)\big\|_2^{\text{pass}} + \big\|\big(I_n \otimes z^{-\ell}\, h_{\text{pre}}(z) - I_n\big)\,\Pi\, u(z)\big\|_2^{\text{stop}}.
\end{aligned}$$

The maximum 2-norm gain between any two signals is given by the ∞-norm of the transfer function connecting them. The ∞-norm of $I_n \otimes z^{-\ell}\, h_{\text{pre}}(z)$ is $\|h_{\text{pre}}(z)\|_\infty$, and the ∞-norm of $I_n \otimes z^{-\ell}\, h_{\text{pre}}(z) - I_n$ is $\|z^{\ell} - h_{\text{pre}}(z)\|_\infty$ These give the bound on the maximum absolute steady-state error,

$$\begin{aligned}
\|e_k\|_\infty \leq \delta \big(&\|h_{\text{pre}}(z)\|_\infty^{\text{pass}}\, \|(I - \Pi)\,u(z)\|_2^{\text{pass}} + \|h_{\text{pre}}(z)\|_\infty^{\text{stop}}\, \|(I - \Pi)\,u(z)\|_2^{\text{stop}}\big) \\
&+ \|z^{-\ell} - h_{\text{pre}}(z)\|_\infty^{\text{pass}}\, \|\Pi\, u(z)\|_2^{\text{pass}} + \|z^{-\ell} - h_{\text{pre}}(z)\|_\infty^{\text{stop}}\, \|\Pi\, u(z)\|_2^{\text{stop}}. \qquad \square
\end{aligned}$$

If the graph satisfies certain properties, then we can bound the term $\|\mathcal{W}_k^\ell - \Pi\|_2$ in (5.28). For example, if the graph is connected, balanced, and has Laplacian matrix which satsfies $\|I - L_k - \Pi_n\|_2 \leq \alpha$ for some $\alpha < 1$ at each iteration, then we have $\|\mathcal{W}_k^\ell - \Pi\|_2 \leq \alpha^\ell$. To prove this, we first need the following lemma.

Lemma 11. *Suppose $\mathcal{G}$ is time-varying and balanced at each iteration, and define $W_k := I_n - L_k$ for $k \geq 0$. Then for all $k \geq \ell$,*

$$W_k W_{k-1} \cdots W_{k-\ell} - \Pi_n = (W_k - \Pi_n)(W_{k-1} - \Pi_n) \cdots (W_{k-\ell} - \Pi_n). \tag{5.30}$$

Proof. We prove the result using induction on ℓ. Note that the result is trivially satisfied when $\ell = 0$. Now assume that the result holds for ℓ, i.e.,

$$W_k W_{k-1} \cdots W_{k-\ell} - \Pi = (W_k - \Pi)(W_{k-1} - \Pi) \cdots (W_{k-\ell} - \Pi).$$

Multiplying both sides by $W_{k+1} - \Pi$,

$$(W_{k+1} - \Pi)\, W_k \cdots W_{k-\ell} - (W_{k+1} - \Pi)\, \Pi = (W_{k+1} - \Pi)(W_k - \Pi) \cdots (W_{k-\ell} - \Pi).$$

We now use the fact that $W_k \Pi = \Pi = \Pi\, W_k$ for all $k \geq 0$ to obtain

$$W_{k+1} W_k \cdots W_{k-\ell} - \Pi = (W_{k+1} - \Pi)(W_k - \Pi) \cdots (W_{k-\ell} - \Pi), \tag{5.31}$$

so the result holds for $\ell + 1$. Therefore, by induction, (5.30) holds for all $\ell \geq 0$. □

Lemma 12. *Suppose the graph Laplacian satisfies $L_k \in \mathbb{L}_n(\alpha)$ for all $k \geq 0$. Then*

$$\|\mathcal{W}_k^\ell - \Pi\|_2 \leq \alpha^\ell \tag{5.32}$$

for all $k \geq \ell$ *where* $\mathcal{W}_k^\ell$ *is defined in* (5.29).

Proof. Since the graph is balanced at each iteration, Lemma 11 implies

$$\mathcal{W}_k^\ell - \Pi = (W_k - \Pi)(W_{k-1} - \Pi) \cdots (W_{k-\ell} - \Pi).$$

Since $L_k \in \mathbb{L}_n(\alpha)$ for all k, $\|W_k - \Pi\|_2 \leq \alpha$ which implies $\|\mathcal{W}_k^\ell - \Pi\|_2 \leq \alpha^\ell$. □

Next, we give conditions under which the steady-state error is arbitrarily small.

Corollary 7. *Consider the estimator in Fig. 5.4d. Suppose the graph* $\mathcal{G}$ *has Laplacian matrix* $L_k \in \mathbb{L}_n(\alpha)$ *for some* $\alpha < 1$ *for all* $k \geq 0$. *Also, suppose the input signals are finite and bandlimited with cutoff frequency* θ_c. *Then the maximum absolute steady-state error can be made arbitrarily small if* $h_{pre}(z)$ *can be chosen such that* σ_{cons}^{pass} *is arbitrarily small for any* ℓ *and exact arithmetic is used.*

Proof. Let $\epsilon > 0$. Since the input signals are bandlimited and exact arithmetic is used, $\|u\|_2^{\text{stop}} = 0$. Since $\alpha < 1$ and the input signals are finite, there exists $\bar{\ell}$ such that $\alpha^\ell \, \|h_{\text{pre}}(z)\|_\infty^{\text{pass}} \, \|(I - \Pi)\, u(z)\|_2^{\text{pass}} < \epsilon/2$ for all $\ell > \bar{\ell}$. Choose ℓ to be an integer greater than $\bar{\ell}$. Then the prefilter can be designed such that $\sigma_{\text{cons}}^{\text{pass}} \, \|\Pi \, u(z)\|_2^{\text{pass}} < \epsilon/2$. Applying Theorem 19 shows that the maximum absolute steady-state error is less than ϵ. □

To summarize, the feedforward estimator in Fig. 5.4d is robust to initial conditions and robust to changes in graphs in $\mathbb{L}_n(\alpha)$. The maximum absolute steady-state error depends on α, the number of stages ℓ, and the prefilter $h_{\text{pre}}(z)$. To obtain small steady-state error, the number of stages ℓ and the prefilter $h_{\text{pre}}(z)$ must be chosen to make $\sigma_{\text{cons}}^{\text{pass}}$ small. In the following section, we design the prefilter to obtain arbitrarily small steady-state error.

5.4. Prefilter Design

The design of the feedforward estimator in Fig. 5.4d requires an ℓ-step bandlimited prediction filter $h_{\text{pre}}(z)$ with the following properties:

- $h_{\text{pre}}(z)$ is proper
- $h_{\text{pre}}(z) \approx z^{\ell}$ for $z = e^{j\theta}$, $\theta \in [0, \theta_c]$.

An ℓ-step filter is obtained by cascading a one-step filter ℓ times in series,

$$h_{\text{pre}}(z) = [zf(z)]^{\ell} \tag{5.33}$$

where $f(z)$ is strictly proper and approximates unity in the passband, i.e.,

- $f(z)$ is strictly proper
- $f(z) \approx 1$ for $z = e^{j\theta}$, $\theta \in [0, \theta_c]$.

Since $f(z)$ must approximate unity in both magnitude *and* phase, a standard lowpass filter cannot be used. Instead, set

$$f(z) = 1 - \frac{g(z)}{\lim_{z\to\infty} g(z)} \tag{5.34}$$

where $g(z)$ is a proper highpass filter with cutoff frequency θ_c, i.e.,

- $g(z)$ is proper
- $g(z) \approx 0$ for $z = e^{j\theta}$, $\theta \in [0, \theta_c]$.

The normalizing constant in the denominator is used to make $f(z)$ strictly proper. Since $g(z)$ is highpass, the magnitude is small in the passband, so $f(z)$ approximates unity in the passband.

Bandlimited prediction filters have been studied extensively [**39, 67, 68, 69, 70**]. It is well known that bandlimited prediction filters are fragile to noise in the inputs [**39**]. Even if the signals are strictly bandlimited, round-off error due to the use of finite precision arithmetic can create high frequency components which are amplified by the filter. However, it has been shown that bandlimited prediction is still possible in the presense of small amounts of noise [**70**]. To account for high-frequency noise, we assume that the frequency spectrum of the input signals is not strictly bandlimited, but has small (nonzero) magnitude for high frequencies ($\theta > \theta_c$). The presence of high-frequency components in the spectrum prohibits the filter from obtaining perfect prediction of bandlimited signals, but we provide bounds for the error and show that it is small when using double precision arithmetic.

The parameters used to characterize the performance of the prefilter which depend on the prefilter are the maximum singular values in (5.27). These can be bounded in terms of $g(z)$ by the following lemma.

Lemma 13. *The following are upper bounds for the maximum singular values:*

$$\begin{aligned} \sigma_{cons}^{pass} \leq \tilde{\sigma}_{cons}^{pass} &:= \left(1 + \frac{\|g\|_\infty^{pass}}{\lim\limits_{z\to\infty} g(z)}\right)^\ell - 1, \qquad \sigma_{dis}^{pass} \leq \tilde{\sigma}_{dis}^{pass} := \left(1 + \frac{\|g\|_\infty^{pass}}{\lim\limits_{z\to\infty} g(z)}\right)^\ell, \\ \sigma_{cons}^{stop} \leq \tilde{\sigma}_{cons}^{stop} &:= \left(1 + \frac{\|g\|_\infty^{stop}}{\lim\limits_{z\to\infty} g(z)}\right)^\ell - 1, \qquad \sigma_{dis}^{stop} \leq \tilde{\sigma}_{dis}^{stop} := \left(1 + \frac{\|g\|_\infty^{stop}}{\lim\limits_{z\to\infty} g(z)}\right)^\ell. \end{aligned} \tag{5.35}$$

Proof. We prove the bounds for $\sigma_{\text{cons}}^{\text{pass}}$ and $\sigma_{\text{dis}}^{\text{pass}}$; the bounds for $\sigma_{\text{cons}}^{\text{stop}}$ and $\sigma_{\text{dis}}^{\text{stop}}$ are similar. We begin with the bound for $\sigma_{\text{cons}}^{\text{pass}}$. Since $h_{\text{pre}}(z) = [zf(z)]^\ell$,

$$\sigma_{\text{cons}}^{\text{pass}} = \max_{\theta\in[0,\theta_c]} \left|e^{j\ell\theta} - h_{\text{pre}}(e^{j\theta})\right| = \max_{\theta\in[0,\theta_c]} \left|1 - [f(e^{j\theta})]^\ell\right| .$$

The binomial expansion of f^ℓ is

$$[f(z)]^\ell = \sum_{k=0}^{\ell} \binom{\ell}{k} (f(z) - 1)^k.$$

Substituting the binomial expansion for $[f(e^{j\theta})]^\ell$ into the expression for $\sigma_{\text{cons}}^{\text{pass}}$,

$$\begin{aligned}
\sigma_{\text{cons}}^{\text{pass}} &= \max_{\theta \in [0,\theta_c]} \left| 1 - \sum_{k=0}^{\ell} \binom{\ell}{k} (f(e^{j\theta}) - 1)^k \right| \\
&= \max_{\theta \in [0,\theta_c]} \left| \sum_{k=1}^{\ell} \binom{\ell}{k} (f(e^{j\theta}) - 1)^k \right| \\
&\leq \max_{\theta \in [0,\theta_c]} \sum_{k=1}^{\ell} \binom{\ell}{k} \left| f(e^{j\theta}) - 1 \right|^k .
\end{aligned}$$

Using (5.34), we have

$$\begin{aligned}
\sigma_{\text{cons}}^{\text{pass}} &\leq \max_{\theta \in [0,\theta_c]} \sum_{k=1}^{\ell} \binom{\ell}{k} \left| \frac{g(e^{j\theta})}{\lim_{z\to\infty} g(z)} \right|^k \\
&= -1 + \sum_{k=0}^{\ell} \binom{\ell}{k} \left(\frac{\|g\|_\infty^{\text{pass}}}{\lim_{z\to\infty} g(z)} \right)^k \\
&= -1 + \left(1 + \frac{\|g\|_\infty^{\text{pass}}}{\lim_{z\to\infty} g(z)} \right)^\ell
\end{aligned}$$

which proves the bound for $\sigma_{\text{cons}}^{\text{pass}}$. To bound $\sigma_{\text{dis}}^{\text{pass}}$, we use

$$\sigma_{\text{dis}}^{\text{pass}} = \|z^\ell - (z^\ell - h_{\text{pre}}(z))\|_\infty^{\text{pass}} \leq \|z^\ell\|_\infty^{\text{pass}} + \|z^\ell - h_{\text{pre}}(z)\|_\infty^{\text{pass}} = 1 + \sigma_{\text{cons}}^{\text{pass}}. \qquad \square$$

Therefore, the maximum absolute steady-state error in (5.26) is minimized by solving the following problem.

Problem 6. *Given the following parameters:*

Parameter	Description
α	the set $\mathbb{L}_n(\alpha)$ of graph Laplacian matrices
ℓ_{max}	maximum number of stages
d_{max}	maximum degree of $g(z)$
$\\|\Pi\, u(z)\\|_2^{pass}$	size of u in the passband and consensus direction
$\\|\Pi\, u(z)\\|_2^{stop}$	size of u in the stopband and consensus direction
$\\|(I-\Pi)\, u(z)\\|_2^{pass}$	size of u in the passband and disagreement directions
$\\|(I-\Pi)\, u(z)\\|_2^{stop}$	size of u in the stopband and disagreement directions

Choose the number of stages ℓ and the filter $g(z)$ to solve the following optimization problem:

$$\begin{aligned} &\min_{\ell,\, g(z)} \; \tilde{\sigma}_{cons}^{pass} \|\Pi\, u(z)\|_2^{pass} + \tilde{\sigma}_{cons}^{stop} \|\Pi\, u(z)\|_2^{stop} \\ &\qquad + \alpha^{\ell} \left(\tilde{\sigma}_{dis}^{pass} \|(I-\Pi)\, u(z)\|_2^{pass} + \tilde{\sigma}_{dis}^{stop} \|(I-\Pi)\, u(z)\|_2^{stop} \right) \\ &\text{subject to} \qquad \ell \le \ell_{max} \\ &\qquad\qquad\qquad deg(g) \le d_{max} \\ &\qquad\qquad\qquad g(z) \;\; proper \end{aligned} \tag{5.36}$$

where $\tilde{\sigma}_{cons}^{pass}$, $\tilde{\sigma}_{cons}^{stop}$, $\tilde{\sigma}_{dis}^{pass}$, and $\tilde{\sigma}_{dis}^{stop}$ are defined in (5.35)*.*

Remark 5. *If the input signal is bandlimited with cutoff frequency θ_c and exact precision arithmetic is used, then $\|u(z)\|_2^{stop} = 0$, so $\|g\|_\infty^{stop}$ can be arbitrarily large. When using finite precision arithmetic, however, we set $\|u(z)\|_2^{stop}$ equal to the machine epsilon. Then $\|g\|_\infty^{stop}$ must be accounted for in the optimization problem* (5.36)*.*

The prefilter can be designed using either a finite impulse response (FIR) or infinite impulse response (IIR) filter for $g(z)$. We now give designs for both types and explain the advantages and disadvantages of both.

5.4.1. FIR Prefilter Design

To design $g(z)$ as an FIR filter, we follow the approach in [**39**]. The filter design is based on the highpass Dolph-Chebyshev window function

$$D_d(\theta) = T_d\left(\frac{\cos((\pi-\theta)/2)}{\cos((\pi-\theta_c)/2)}\right) \tag{5.37}$$

where T_d is the degree d Chebyshev polynomial of the first kind. The unique feature of $D_d(\theta)$ is that it oscillates between 1 and -1 for $\theta \in [-\theta_c, \theta_c]$, as shown in Fig. 5.5.

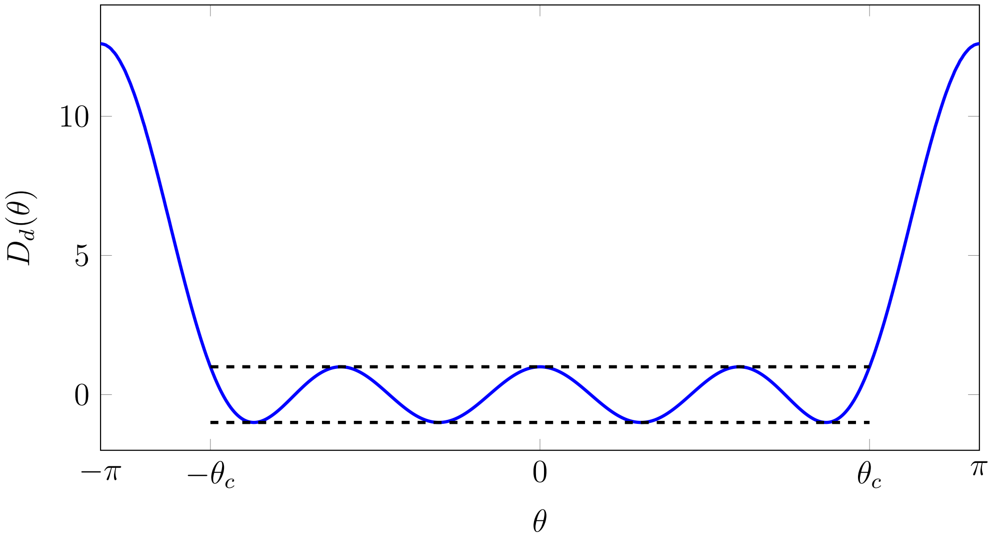


Figure 5.5. Plot of the Dolph-Chebyshev window function (5.37) with $d = 8$ and $\theta_c = 3\pi/4$. The horizontal dashed lines are at ±1.

The FIR filter is designed such that the magnitude of the frequency response is given by $|D_d(\theta)|$. The filter is summarized in the following lemma.

Lemma 14. *Let $g(z)$ be such that*

$$g(-z) = z^{-d/2}\, T_d\left(\frac{(z^{1/2} + z^{-1/2})/2}{\cos\left((\pi - \theta_c)/2\right)}\right). \tag{5.38}$$

Then the following hold:

(1) $g(z)$ is a proper FIR transfer function of degree d,

(2) $\|g\|_\infty^{pass} = 1$,

(3) $\|g\|_\infty^{stop} = T_d(1/\cos((\pi - \theta_c)/2))$, and

(4) $\lim_{z\to\infty} g(z) = 1/(2\cos^d((\pi - \theta_c)/2))$.

Proof. Since $T_d(x)$ is the degree d Chebyshev polynomial of the first kind, it is composed of all even powers of x if d is even and all odd powers of x if d is odd. Then

$$g(-z) = z^{-d/2}\, T_d\left(\frac{(z^{1/2} + z^{-1/2})/2}{\cos\left((\pi - \theta_c)/2\right)}\right) = \sum_{k=0}^{d} c_k\, z^{-k}$$

so $g(z)$ is a proper FIR transfer function of degree d.

The leading coefficient of $T_d(x)$ is 2^{d-1}, so

$$\lim_{z\to\infty} g(z) = \lim_{z\to\infty} z^{-d/2}\, 2^{d-1} \left(\frac{(z^{1/2} + z^{-1/2})/2}{\cos((\pi - \theta_c)/2)}\right)^d = \frac{1}{2\cos^d((\pi - \theta_c)/2)}.$$

Using the trigonometric identity $\cos((\pi - \theta)/2) = ((-e^{j\theta})^{1/2} + (-e^{j\theta})^{-1/2})/2$, the magnitude of $g(z)$ evaluated on the unit circle is

$$|g(e^{j\theta})| = \left| T_d\left(\frac{\cos((\pi - \theta)/2)}{\cos((\pi - \theta_c)/2)}\right)\right|.$$

The Chebyshev polynomial $T_d(x)$ oscillates between 1 and -1 for $x \in [-1, 1]$, and the magnitude increases monotonically in x for $|x| > 1$. For $\theta \in [0, \theta_c]$, the argument of T_d is in $[-1,1]$, so $\|g(z)\|_\infty^{\text{pass}} = 1$. For $\theta \in [\theta_c, \pi]$, the argument of T_d is outside of $[-1,1]$, so the maximum value occurs when the argument is largest. This occurs at $\theta = \pi$ which gives $\|g(z)\|_\infty^{\text{stop}} = T_d(1/\cos((\pi-\theta_c)/2))$. □

The next lemma shows that this choice of $g(z)$ can be used to make $\sigma_{\text{cons}}^{\text{pass}}$ arbitrarily small.

Lemma 15. *Consider the FIR prefilter given by* (5.38)*. Let* $\theta_c \in [0,\pi)$*. Then* σ_{cons}^{pass} *in* (5.27) *can be made arbitrarily small. That is,*

$$\lim_{\substack{\ell\to\infty \\ d\to\infty}} \sigma_{cons}^{pass} = 0. \tag{5.39}$$

Proof (sketch): The result is obtained by substituting the values in Lemma 14 into the upper bound for $\sigma_{\text{cons}}^{\text{pass}}$ in (5.35) and taking the limit as $\ell \to \infty$ and $d \to \infty$. □

Combining this result with Corollary 7 shows that the FIR prefilter can be used with the feedforward estimator to obtain arbitrarily small steady-state error when exact arithmetic is used. The error cannot be made arbitrarily small when using finite precision arithmetic, but it can be minimized by solving Problem 6. The optimization problem can be solved by evaluating the value of the objective function in (5.36) for $\ell = 1, 2, \ldots, \ell_{\max}$ and $d = 1, 2, \ldots, d_{\max}$ and choosing the parameters which obtain the smallest value for the objective function.

5.4.2. IIR Prefilter Design

An infinite impulse response design for the prefilter is obtained by setting $g(z)$ to be a Type II Chebyshev highpass filter as described in the following lemma.

Lemma 16. *Let $g(z)$ be a Type II Chebyshev highpass filter of degree d with stopband attenuation parameter ϵ and cutoff frequency θ_c, i.e.,*

$$g(z) = K \prod_{i=1}^{d} \frac{z - z_i}{z - p_i} \tag{5.40}$$

where $K = \prod_{i=1}^{d}(1 + p_i)/(1 + z_i)$ and

$$z_i = \frac{2 + j\omega_0 \cos(\theta_i)}{2 - j\omega_0 \cos(\theta_i)}, \qquad p_i = \frac{2 + j\omega_0 \cos(\theta_i - j\gamma)}{2 - j\omega_0 \cos(\theta_i - j\gamma)}, \tag{5.41}$$

with $\omega_0 = 2\tan(\theta_c/2)$, $\gamma = \operatorname{asinh}(1/\epsilon)/d$, and $\theta_i = \pi(2i - 1)/(2d)$ for $i = 1, \ldots, d$. Then the following hold:

(1) $g(z)$ is a proper IIR transfer function of degree d,

(2) $\|g\|_\infty^{pass} = \epsilon/\sqrt{1 + \epsilon^2}$,

(3) $\|g\|_\infty^{stop} = 1$, and

(4) $\lim_{z\to\infty} g(z) = K$.

Proof. The transfer function of the Type II Chebyshev highpass filter is obtained as follows. First, the corresponding continuous filter $G(s)$ is obtained. The continuous filter is the unique stable filter with frequency response

$$|G(j\omega)| = \frac{\epsilon\, T_d(\omega/\omega_0)}{\sqrt{1 + \epsilon^2\, T_d^2(\omega/\omega_0)}} \tag{5.42}$$

where T_d is the degree d Chebyshev polynomial of the first-kind, ω_0 is the cutoff frequency, and ϵ is the stopband attenuation parameter. The s-plane poles are the left-half plane roots of the denominator of (5.42), and the zeros are the roots of the numerator with multiplicity one. These are given by $P_i = j\omega_0 \cos(\theta_i - j\gamma)$ and $Z_i = j\omega_0 \cos(\theta_i)$ for $i = 1, \ldots, d$ where

$\theta_i = \pi(2i-1)/(2d)$, $\gamma = \operatorname{asinh}(1/\epsilon)/d$, and $j = \sqrt{-1}$ is the imaginary unit. The z-plane poles and zeros are then obtained using the bilinear transform, $p_i = (2+P_i)/(2-P_i)$ and $z_i = (2+Z_i)/(2-Z_i)$. The gain is chosen such that the transfer function is unity at $z=-1$, so $K = \prod_{i=1}^{d}(1+z_i)/(1+p_i)$. Since the bilinear transform warps the frequencies, we take $\omega_0 = 2\tan(\theta_c/2)$ so that the cutoff frequency of the discrete filter is θ_c. The transfer function of the discrete Type II Chebyshev highpass filter is then given by (5.40), so $g(z)$ is a proper IIR transfer function of degree d. The magnitude of the filter response oscillates between 0 and $\epsilon/\sqrt{1+\epsilon^2}$ in the stopband, so

$$\|g\|_\infty^{\text{pass}} = \max_{\theta\in[-\theta_c,\theta_c]} \left|g(e^{j\theta})\right| = \frac{\epsilon}{\sqrt{1+\epsilon^2}}. \tag{5.43}$$

The magnitude of the frequency response approaches unity at $\theta=\pi$, so $\|g\|_\infty^{\text{stop}} = 1$. Also, from (5.40) we have that $\lim_{z\to\infty} g(z) = K$. □

Example Bode plots of $g(z)$ and $f(z)$ are given in Fig. 5.6. Note that $f(z)$ is strictly proper and approximates unity in the passband.

Similar to the FIR filter, this choice of $g(z)$ can also be used to make $\sigma_{\text{cons}}^{\text{pass}}$ arbitrarily small. Before presenting the result, we need the following lemma which characterizes the gain K in the limit as $d\to\infty$.

Lemma 17. *In the limit as the degree of the filter approaches infinity, the filter gain approaches*

$$\lim_{d\to\infty} K = \exp\left\{-\frac{\theta_c}{\pi}\sinh^{-1}\left(\frac{1}{\epsilon}\right)\right\}. \tag{5.44}$$

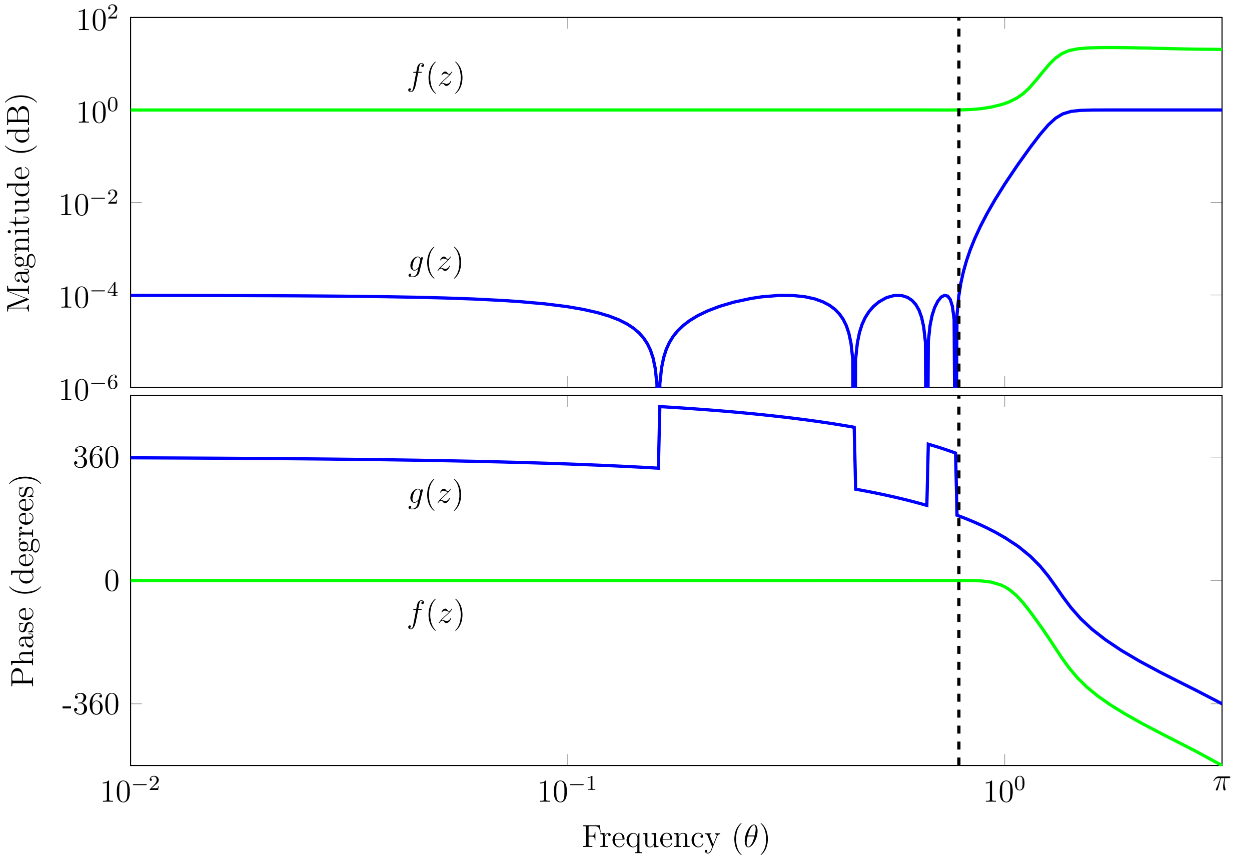


Figure 5.6. Bode plot of the prefilter. Shown are $g(z)$ (blue) and $f(z)$ (green) with $d = 8$, $\epsilon = 10^{-4}$, and $\theta_c = \pi/4$. The vertical line indicates θ_c.

Proof. Let $\theta_k = \frac{\pi}{2d}(2k-1)$ for $k = 1, \ldots, d$, and let $\gamma = \sinh^{-1}(1/\epsilon)/d$. The pre-warped cutoff frequency is $\omega_0 = 2\tan(\theta_c/2)$. The filter gain is then

$$K = \prod_{k=1}^{d} \frac{2 - j\omega_0 \cos(\theta_k)}{2 - j\omega_0 \cos(\theta_k - j\gamma)} = \prod_{k=1}^{d} \frac{2 - j\omega_0 \cos(\theta_k)}{2 - j\omega_0 \cos(\theta_k - j\frac{1}{\pi}\sinh^{-1}(1/\epsilon)d\theta)}$$

where $d\theta = \theta_{k+1} - \theta_k = \pi/d$. Taking the limit as $d \to \infty$ gives an infinite product. To evaluate the product, we take the logarithm of K which instead gives an infinite sum,

$$\begin{aligned}\lim_{d\to\infty} \ln(K) &= \int_0^\pi \ln\left(\frac{2 - j\omega_0\cos(\theta)}{2 - j\omega_0\cos(\theta - j\frac{1}{\pi}\sinh^{-1}(1/\epsilon)d\theta)}\right) \\ &= \int_0^\pi \ln\left(2 - j\omega_0\cos(\theta)\right) - \ln\left(2 - j\omega_0\cos(\theta - j\frac{1}{\pi}\sinh^{-1}(1/\epsilon)d\theta)\right).\end{aligned}$$

The integral cannot be evaluated directly since $d\theta$ is inside the function. Since $d\theta$ is infitesimally small, we expand the second term about $d\theta = 0$,

$$\begin{aligned}\ln\left(2 - j\omega_0\cos(\theta - j\frac{1}{\pi}\sinh^{-1}(1/\epsilon)d\theta)\right) = \ln\left(2 - j\omega_0\cos(\theta)\right) + \frac{\omega_0}{\pi}\frac{\sinh^{-1}(1/\epsilon)\sin(\theta)}{2 - j\omega_0\cos(\theta)}d\theta \\ +\mathcal{O}(d\theta^2).\end{aligned}$$

Using this expansion,

$$\lim_{d\to\infty} \ln(K) = \int_0^\pi -\frac{\omega_0}{\pi}\frac{\sinh^{-1}(1/\epsilon)\sin(\theta)}{2 - j\omega_0\cos(\theta)}d\theta + \mathcal{O}(d\theta^2).$$

The higher order terms in $d\theta$ can be neglected, so

$$\lim_{d\to\infty} \ln(K) = -\frac{\omega_0}{\pi}\sinh^{-1}(1/\epsilon)\int_0^\pi \frac{\sin(\theta)}{2 - j\omega_0\cos(\theta)}d\theta.$$

The complex integrand is symmetric about $\theta = \pi/2$, and can therefore be computed using the real integral

$$\lim_{d\to\infty} \ln(K) = -\frac{\omega_0}{\pi}\sinh^{-1}(1/\epsilon)\int_0^{\pi/2} \frac{4\sin(\theta)}{4 + \omega_0^2\cos^2(\theta)}d\theta = -\frac{2}{\pi}\sinh^{-1}(1/\epsilon)\tan^{-1}(\omega_0/2).$$

Using the fact that $\omega_0 = 2\tan(\theta_c/2)$, this simplifies to

$$\lim_{d\to\infty} \ln(K) = -\frac{\theta_c}{\pi}\sinh^{-1}(1/\epsilon). \qquad \square$$

Lemma 18. *Consider the IIR prefilter given by* (5.40)*. Let $\theta_c \in [0, \pi)$. Then σ^{pass}_{cons} in* (5.27) *can be made arbitrarily small. That is,*

$$\lim_{\substack{\ell\to\infty \\ d\to\infty \\ \epsilon\to 0^+}} \sigma^{pass}_{cons} = 0. \tag{5.45}$$

Proof (sketch): The result is obtained by substituting the value in Lemma 16 into the upper bound for $\sigma^{\text{pass}}_{\text{cons}}$ in (5.35) and taking the limit as $\ell \to \infty$, $d \to \infty$, and $\epsilon \to 0^+$. □

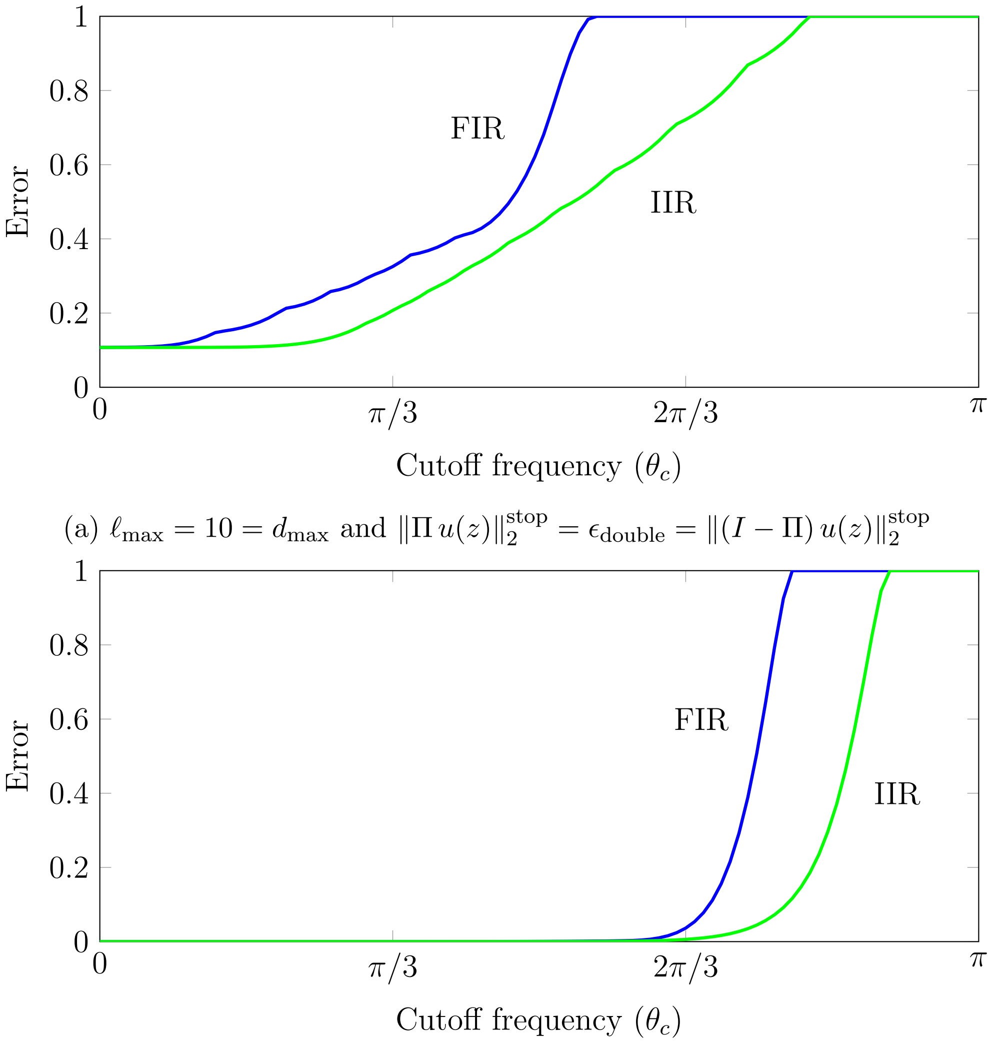
Combining this result with Corollary 7 shows that the IIR prefilter can be used with the feedforward estimator to obtain arbitrarily small steady-state error when exact arithmetic is used. The error cannot be made arbitrarily small when using finite precision arithmetic, but it can be minimized by solving Problem 6. The optimization problem can be solved as follows. First, note that K is an increasing function of d, and

$$\begin{aligned} \tilde{\sigma}^{\text{pass}}_{\text{cons}} &= \left(1 + \frac{1}{K}\frac{\epsilon}{\sqrt{1+\epsilon^2}}\right)^{\ell} - 1, & \tilde{\sigma}^{\text{pass}}_{\text{dis}} &= \left(1 + \frac{1}{K}\frac{\epsilon}{\sqrt{1+\epsilon^2}}\right)^{\ell} \\ \tilde{\sigma}^{\text{stop}}_{\text{cons}} &= \left(1 + \frac{1}{K}\right)^{\ell} - 1, & \tilde{\sigma}^{\text{stop}}_{\text{dis}} &= \left(1 + \frac{1}{K}\right)^{\ell}. \end{aligned} \tag{5.46}$$

We want to maximize K to minimize all of the maximum singular values, so the largest value of d should be used which is $d = d_{\max}$. Now consider solving the problem for fixed ℓ. Then ℓ and d are known so we can perform a bisection search on ϵ to find the value that minimizes the objective. This is performed for $\ell = 1, 2, \ldots, \ell_{\max}$ and the solution taken to be the parameters which give the smallest value for the objective.

5.4.3. Prefilter Comparison (FIR vs. IIR)

We have presented both FIR and IIR designs for the prefilter $h_{\text{pre}}(z)$. We now compare the two choices for the prefilter.



(a) $\ell_{\max} = 10 = d_{\max}$ and $\|\Pi\, u(z)\|_2^{\text{stop}} = \epsilon_{\text{double}} = \|(I - \Pi)\, u(z)\|_2^{\text{stop}}$

(b) $\ell_{\max} = 50 = d_{\max}$ and $\|\Pi\, u(z)\|_2^{\text{stop}} = 0 = \|(I - \Pi)\, u(z)\|_2^{\text{stop}}$

Figure 5.7. Plot of the solution to Problem 6 for the FIR (blue) and IIR (green) prefilters in Sections 5.4.1 and 5.4.2, respectively. The parameters used are $\alpha = 0.8$ and $\|\Pi\, u(z)\|_2^{\text{pass}} = 1 = \|(I - \Pi)\, u(z)\|_2^{\text{pass}}$. The machine epsilon for double precision is $\epsilon_{\text{double}} = 2.22 \times 10^{-16}$.

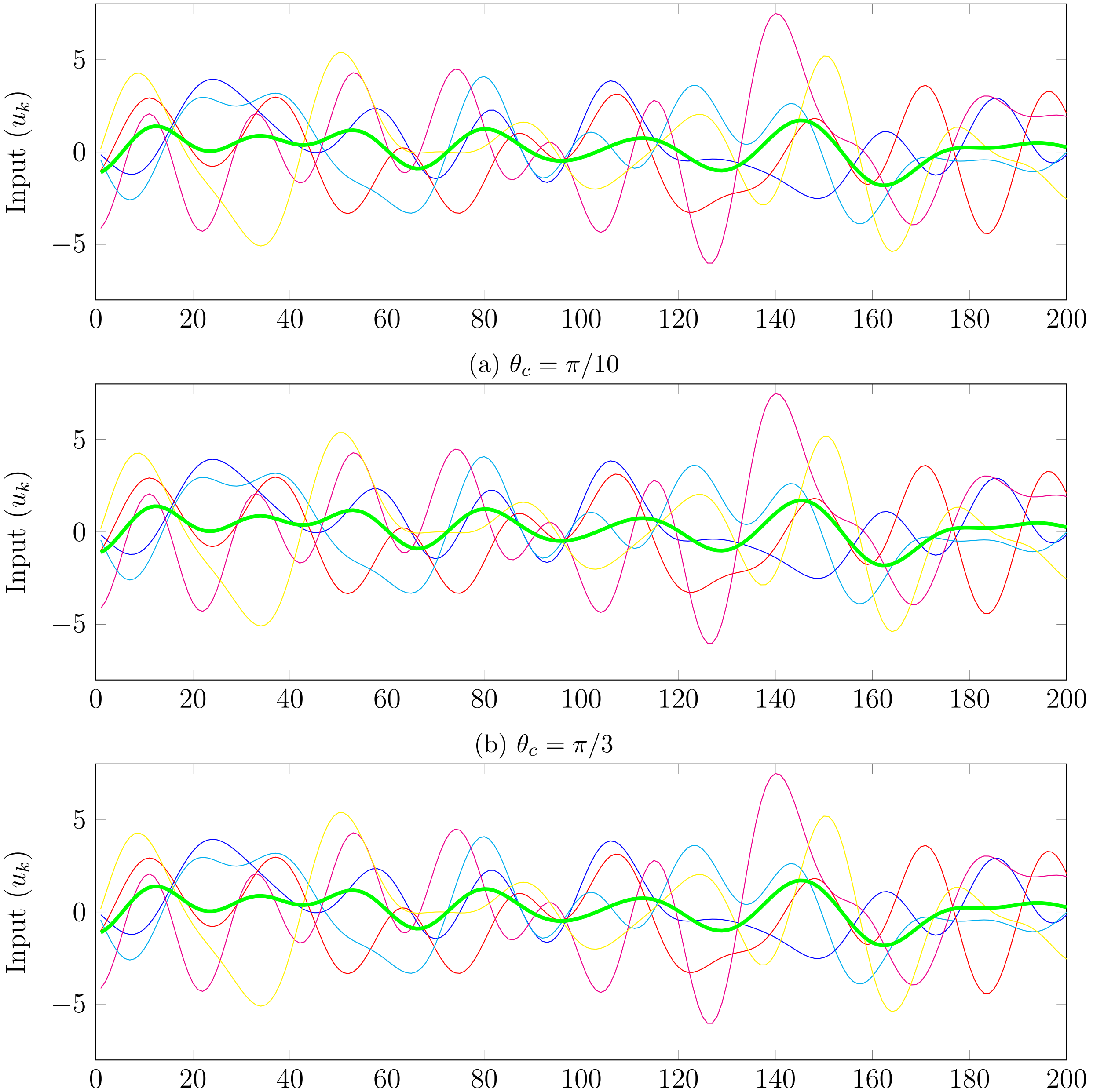
The FIR design is desirable due to its simplicity and finite convergence properties. The filter is simple to implement since the output is a linear combination of the previous input values. In addition, all of the poles of the filter are at zero, so the filter converges in a finite number of iterations equal to its degree. Therefore, the convergence of the FIR filter is in general much faster than the IIR filter. The maximum absolute steady-state error, however, is in general smaller for the IIR filter as shown in Fig. 5.7. Lemmas 15 and 18 along with Corollary 7 indicate that the maximum absolute steady-state errors for both the FIR and IIR filters can be made arbitrarily small if the signal is bandlimited and exact arithmetic is used. When this is not the case, however, the error of the IIR filter is smaller than the FIR filter. Therefore, the IIR filter can achieve smaller error in practical applications.

Remark 6. *The maximum absolute steady-state error never goes above $\|u(z)\|_2$ in Fig. 5.7. When $\ell = 0$ and $h_{pre}(z) = 1$ (so the output is simply equal to the input), the maximum absolute steady-state error is bounded by $\|u(z)\|_2$.*

5.5. Simulations

In this section, we verify the estimator designs by simulating them on various communication graphs and input signals. We use both constant and time-varying graphs for the simulations.

We construct the input signals to be bandlimited as follows. First, we assign random complex values to the frequency spectrum $[0, \theta_c]$ of the Fourier transform of each input signal. The complex conjugate values are then assigned to $[-\theta_c, 0]$ so that the input signal is real, and the spectrum $[-\pi, -\theta_c] \cup [\theta_c, \pi]$ is set to zero. We then take the inverse Fourier transform to produce the input signal. Next, we normalize the input signals. Let $\bar{u}_k = (1/n) \sum_{i=1}^{n} u_k^i$ be the average of the input signals at time k. The input signals are then all shifted by a

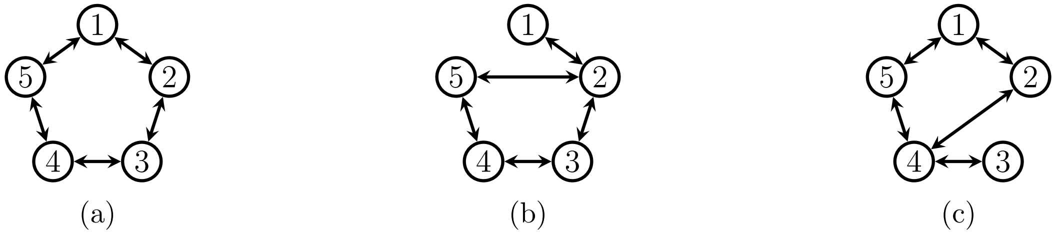


(a) $\theta_c = \pi/10$

(b) $\theta_c = \pi/3$

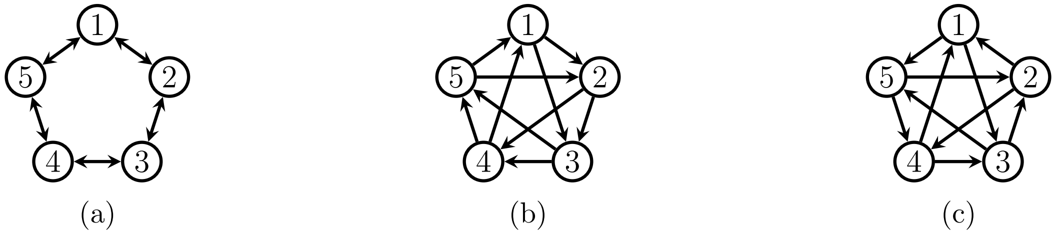
(c) $\theta_c = 2\pi/3$

Figure 5.8. Example input signals used for simulations. Plotted are the input signals u_k^i (thin lines) and the exact average of the inputs $\mathbf{1}_n^T u_k/n$ (thick green) as a function of the iteration k.



(a) (b) (c)

Figure 5.9. Undirected graphs used in the simulations. The nonzero weights are all chosen to be 0.3766 so that $L \in \mathbb{L}_n(0.6875)$ for each graph Laplacian.



(a) (b) (c)

Figure 5.10. Directed graphs used in the simulations. There is an arrow from i to j if agent i receives information from agent j, i.e., $(i, j) \in \mathcal{E}$. The nonzero weights are all chosen to be 0.3333 so that $L \in \mathbb{L}_n(0.5393)$ for each graph Laplacian.

constant so that the mean of $\bar{u}_k$ is zero, and then scaled by a constant so that $\bar{u}_k$ has unit standard deviation. Example input signals for different values of θ_c are shown in Fig. 5.8.

The undirected and directed communication graphs used in the simulations are shown in Figs. 5.9 and 5.10, respectively. All of the graphs are strongly connected and balanced. For each set of graphs, the nonzero weights are all the same value. The value of the nonzero weights is chosen to minimize α such that the graph Laplacian is in $\mathbb{L}_n(\alpha)$ for each graph in the set. These values are indicated in Figs. 5.9 and 5.10.

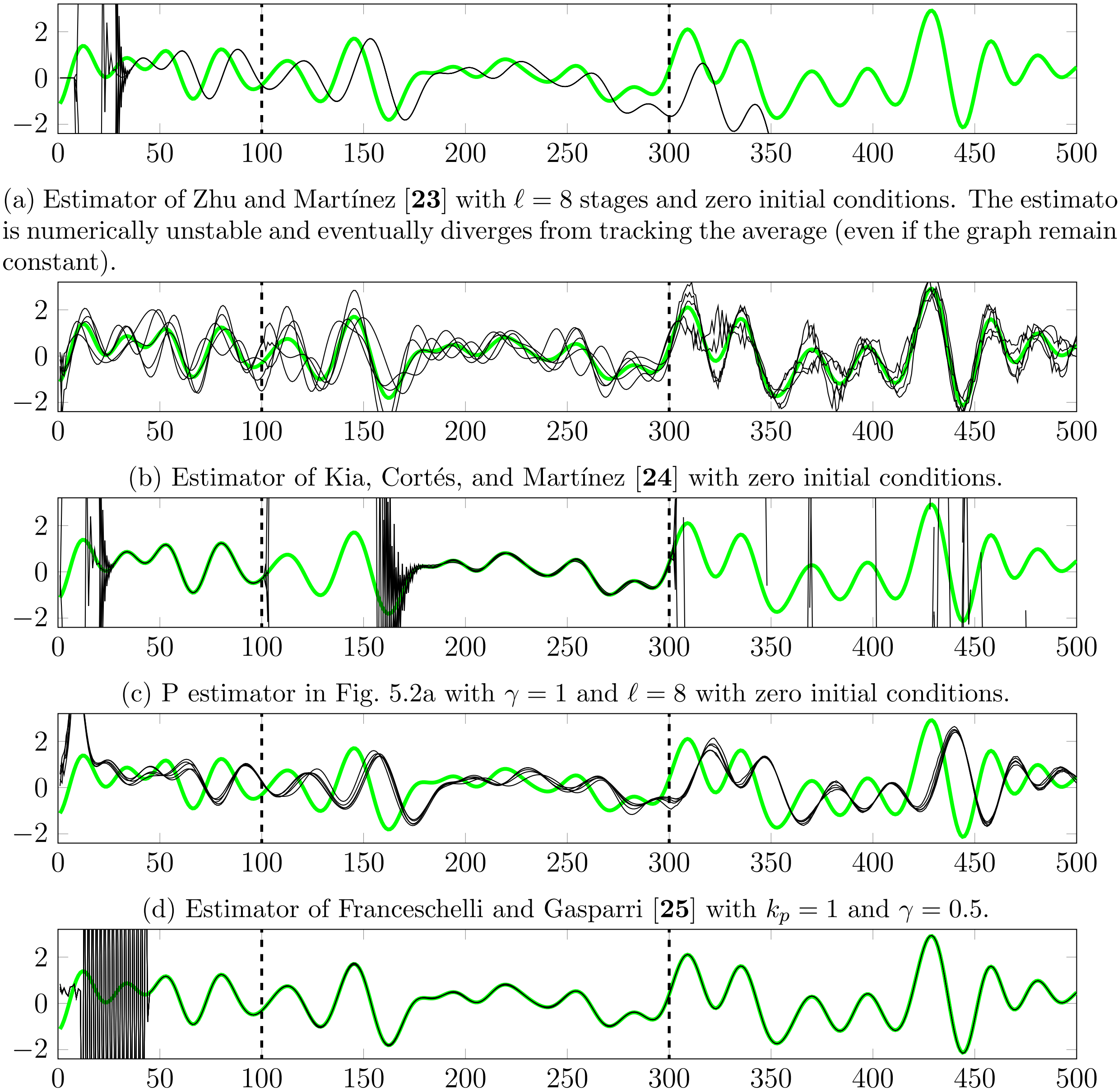
The first feedback estimator considered is that of Zhu and Martínez [**23**]. Since the estimator is not robust to initial conditions, the estimator states are all initialized to zero. As shown in Fig. 5.11a, the estimator tracks the average of the inputs, but with a delay equal to the number of cascade stages ℓ. Also, the estimator is numerically unstable. The prefilter is the ℓ^{th} order divided difference of the input signal which is then passed through ℓ integrators. Differentiation amplifies high frequencies causing this to be unstable, even for small values of ℓ.

The estimator of Kia, Cortés, and Martínez [**24**] is simulated in Fig. 5.11b. After a pole/zero cancellation, this is equivalent to the P estimator with $\gamma = 1$ and $\ell = 1$. The estimator fixes the delay and stability issues, although it is still not robust to initial conditions, and the error is large with only a single stage.

More stages can be used to reduce the error as shown in Fig. 5.11c. Using $\ell = 8$ reduces the error when the graph is constant, but also makes the estimator non-robust to changes in the graph.

A simulation of the estimator of Franceschelli and Gasparri [**25**] is shown in Fig. 5.11d. This estimator is both robust to initial conditions and changes in the graph. However, the error cannot be made arbitrarily small since the consensus direction is not unity, i.e., $G(z,0) \neq 1$. Instead of the outputs approximating the average of the input signals, the outputs approximate the average of the input signals applied to the filter $G(z,0) = 1/(z-\gamma)^{\ell}$. A prefilter could be used to reduce the error in the consensus direction as in Fig. 5.2b. However, Theorem 18 shows that, even using the ideal prefilter, the estimator is not able to achieve arbitrarily small error for bandlimited input signals with $\theta_c < \pi$ over time-varying graphs.

An estimator that is capable of achieving arbitrarily small steady-state error and is both robust to initial conditions and changes in the graph is the feedforward estimator. The simulation for this estimator using the FIR prefilter is shown in Fig. 5.11e. Using $\alpha = 0.6875$, $\|\Pi\, u(z)\|_2^{\text{pass}} = 1 = \|(I-\Pi)\, u(z)\|_2^{\text{pass}}$, $\|\Pi\, u(z)\|_2^{\text{stop}} = \epsilon_{\text{double}} = \|(I-\Pi)\, u(z)\|_2^{\text{stop}}$, $d_{\max} = 4$,



(a) Estimator of Zhu and Martínez [**23**] with $\ell = 8$ stages and zero initial conditions. The estimator is numerically unstable and eventually diverges from tracking the average (even if the graph remains constant).

(b) Estimator of Kia, Cortés, and Martínez [**24**] with zero initial conditions.

(c) P estimator in Fig. 5.2a with $\gamma = 1$ and $\ell = 8$ with zero initial conditions.

(d) Estimator of Franceschelli and Gasparri [**25**] with $k_p = 1$ and $\gamma = 0.5$.

(e) Feedforward estimator in Fig. 5.4d with $\ell = 10$ stages and the FIR prefilter in (5.47).

Figure 5.11. Simulations of different estimators using the input signals with cutoff frequency $\theta_c = \pi/10$ in Fig. 5.8 and the undirected graphs in Fig. 5.9. The graph changes at iteration 100 and every iteration past 300, and is constant otherwise. Plotted are the exact average of the inputs $\mathbf{1}_n^T u_k/n$ (green) and the outputs y_k^i (black) as a function of the iteration k.

$\ell_{\max} = 10$, and $\theta_c = \pi/10$, the FIR prefilter is

$$h_{\text{pre}}(z) = \left(\frac{3.902z^3 - 5.805z^2 + 3.902z - 1}{z^3}\right)^{10} \tag{5.47}$$

and the IIR prefilter is

$$h_{\text{pre}}(z) = \left(\frac{4.411z(z-0.363)(z^2-0.824z+0.611)}{(z^2+0.210z+0.050)(z^2+0.300z+0.458)}\right)^{10}. \tag{5.48}$$

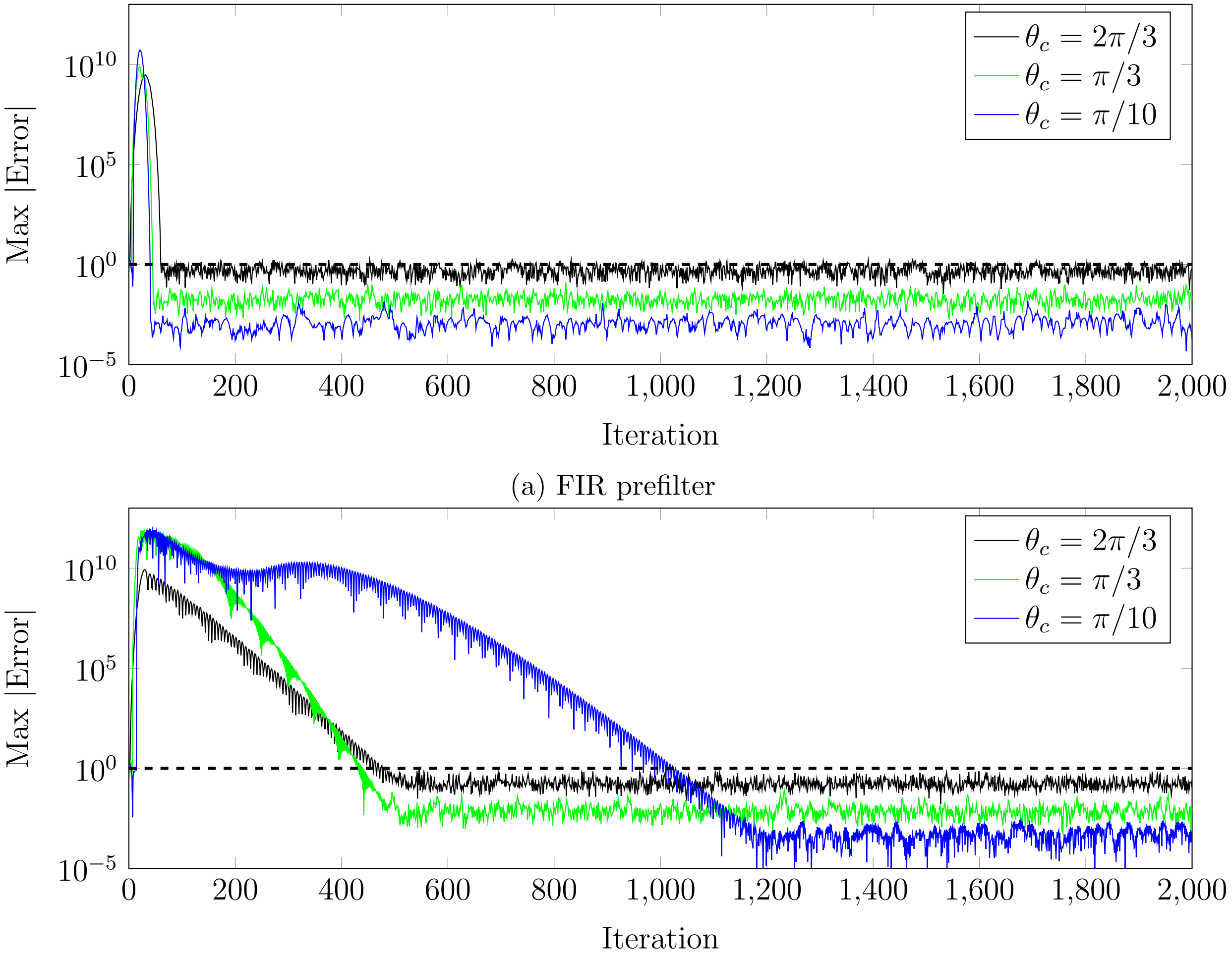
The parameters for both filters are given in Table 5.3.

Table 5.3. Bounds on the maximum absolute steady-state error and maximum singular values of the FIR and IIR prefilters.

	Error	$\tilde{\sigma}^{\text{pass}}_{\text{cons}}$	$\tilde{\sigma}^{\text{pass}}_{\text{dis}}$	$\tilde{\sigma}^{\text{stop}}_{\text{cons}}$	$\tilde{\sigma}^{\text{stop}}_{\text{dis}}$
FIR filter in (5.47)	0.0363	0.0120	0.0239	1.6×10^{12}	3.8×10^{10}
IIR filter in (5.48)	0.0297	0.0056	0.0237	2.1×10^{12}	4.8×10^{10}

5.5.1. Comparison of prefilters

The feedforward estimator in Fig. 5.4d uses the prefilter $h_{\text{pre}}(z)$. Two choices for the prefilter are the FIR and the IIR designs in Sections 5.4.1 and 5.4.2, respectively. To compare the filters, both designs are simulated using the directed graphs in Fig. 5.10 where the graph is chosen randomly at each iteration. The input signals are generated with different cutoff frequencies θ_c. The simulations are shown in Fig. 5.12. The FIR filter converges significantly faster than the IIR filter, although the error is slightly smaller for the IIR filter. The steady-state error and the settling time of both filters increases as θ_c increases.



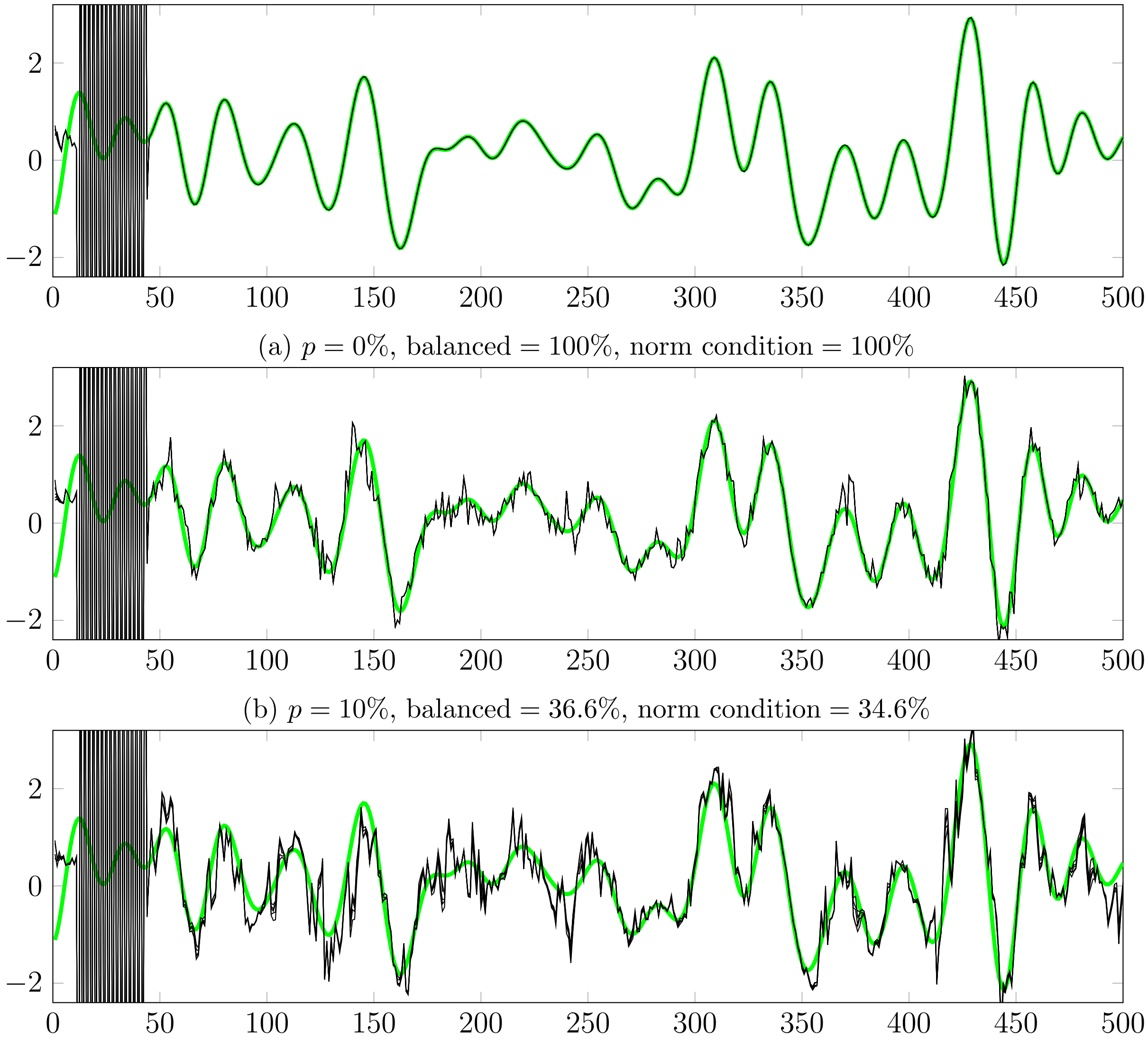
(a) FIR prefilter

(b) IIR prefilter

Figure 5.12. Simulation of the feedforward estimator in Fig. 5.4d using the FIR and IIR prefilter designs. The dashed black line is at unity which is the standard deviation of the input signals.

5.5.2. Dropped packets

So far, it has been assumed that the graph is in the set $\mathbb{L}_n(\alpha)$ at each iteration, i.e., the graph is balanced and satisfies $\|I - L_k - \Pi\|_2 \leq \alpha$. These assumptions may not always hold in practical scenarios. To analyze the proposed feedforward estimator design in such scenarios, the algorithm is implemented on a graph where packets are dropped randomly at



(a) $p = 0\%$, balanced = 100%, norm condition = 100%

(b) $p = 10\%$, balanced = 36.6%, norm condition = 34.6%

(c) $p = 50\%$, balanced = 2.6%, norm condition = 0.2%

Figure 5.13. Simulation of the feedforward estimator in Fig. 5.4d with the FIR prefilter designed using $\alpha = 0.5393$. The input signals and graph used are in Figs. 5.8a and 5.9a, respectively. Packets are dropped randomly at each iteration with probability p. The percent iterations that the graph is balanced and satisfies the norm condition $\|I - L_k - \Pi\|_2 \leq \alpha$ are indicated. Plotted are the exact average of the inputs $\mathbf{1}_n^T u_k / n$ (green) and the outputs y_k^i (black) as a function of the iteration k.

each iteration with probability p. The results are shown in Fig. 5.13. When no packets are dropped ($p = 0$), the estimator tracks the average of the inputs with small error. As the probability of dropping packets increases, the assumptions on the graph are violated more often. When $p = 0.5$, the graph is balanced 2.6% of the iterations and satisfies the norm condition 0.2% of the iterations. As more assumptions are violated, the error increases, but the increase in error is small compared to the number of times the assumptions are violated.

5.6. Summary

We have designed estimators to solve the dynamic average consensus problem when the frequency spectrum of the input signals is composed of a continuous band of frequencies. First, we showed that the feedback estimators from Chapter 4 are incapable of solving Problem 5. Then we designed a feedforward estimator which solves the problem with arbitrarily small steady-state error over time-varying graphs. The only assumptions made on the communication graph are that on average it is connected, balanced, and has Laplacian matrix in $\mathbb{L}_n(\alpha)$ for some $\alpha < 1$.

The estimator was designed for bandlimited input signals with known cutoff frequency $\theta_c < \pi$. This design can be extended to more general frequency bands of the form $[\theta_1, \theta_2]$ by modifying the prefilter to approximate unity in the frequency band. This could also be extended to the case when the magnitude of the frequency spectrum is known (or approximately known) over all frequencies. In other words, the feedforward estimator along with an appropriate prefilter could track input signals with minimal error when an upper bound on $|u(e^{j\theta})|$ is known for all $\theta \in [0, \pi]$.

CHAPTER 6

Convex Optimization

In contrast to the dynamic average consensus problem which is distributed, in this chapter we analyze and design a novel gradient-based algorithm for the unconstrained minimization of strongly convex functions in the centralized setting. When the objective function is m-strongly convex and its gradient is L-Lipschitz continuous, the iterates and function values converge linearly to the optimum at rates ρ and ρ^2, respectively, where $\rho = 1-\sqrt{m/L}$. These are the fastest known guaranteed linear convergence rates for globally convergent first-order methods, and for high desired accuracies the corresponding iteration complexity is within a factor of two of the theoretical lower bound. We use a simple graphical design procedure based on integral quadratic constraints to derive closed-form expressions for the algorithm parameters. The new algorithm, which we call the triple momentum method, can be seen as an extension of methods such as gradient descent, Nesterov's accelerated gradient descent, and the heavy-ball method.

6.1. Summary of Methods

Consider the optimization problem

$$\underset{x\in\mathbb{R}^n}{\text{minimize}} \;\; f(x) \tag{6.1}$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable, strongly convex with parameter m, and has a Lipschitz continuous gradient with Lipschitz constant L. This class of functions is characterized as follows.

Definition 21 (function class)**.** *For a given $0 < m \leq L$, define $\mathcal{S}_{m,L}$ to be the set of functions $f : \mathbb{R}^n \to \mathbb{R}$ that are continuously differentiable, strongly convex with parameter m, and have Lipschitz gradients with Lipschitz constant L. Furthermore, $\kappa := L/m$ is called the* condition number *of $f \in \mathcal{S}_{m,L}$.*

As we will see in Lemma 23, an equivalent characterization for functions $f \in \mathcal{S}_{m,L}$ is that

$$0 \geq \left[\nabla f(x) - \nabla f(y) - m\,(x-y)\right]^T \left[\nabla f(x) - \nabla f(y) - L\,(x-y)\right] \quad \text{for all } x, y \in \mathbb{R}^n. \tag{6.2}$$

The function $f \in \mathcal{S}_{m,L}$ with $m > 0$ is strongly convex and therefore has a unique global minimizer $x_\star \in \mathbb{R}^n$. We consider gradient-based algorithms for solving problem (6.1) of the form

$$\begin{aligned} \xi_{k+1} &= (1+\beta)\xi_k - \beta\xi_{k-1} - \alpha\nabla f(y_k) \\ y_k &= (1+\gamma)\xi_k - \gamma\xi_{k-1} \\ x_k &= (1+\delta)\xi_k - \delta\xi_{k-1} \end{aligned} \tag{6.3}$$

where $\xi \in \ell_{2e}^n$ is the internal state, the gradient is applied to $y \in \ell_{2e}^n$, the output is $x \in \ell_{2e}^n$, and $\xi_0, \xi_{-1} \in \mathbb{R}^n$ are the initial conditions. In this paper we assume the parameters α, β, γ, and δ are constant (i.e., they do not change with k). Table 6.1 shows how some known methods are of the form (6.3) with particular constraints on these parameters.

Given bounds on the distance of the iterates from the optimizer, we can obtain bounds on the function values using Lemma 19.

Table 6.1. Parameters of gradient-based optimization algorithms (up to a change of variables).

Method	Parameters $(\alpha, \beta, \gamma, \delta)$
Gradient descent	$(\alpha, 0, 0, 0)$
Heavy-ball method [**47, 50**]	$(\alpha, \beta, 0, 0)$
Nesterov's accelerated gradient descent [**48**]	$(\alpha, \beta, \beta, 0)$
Regularized update descent [**51**]	$(\alpha, \beta, 1, 0)$
Algorithm in [**49**, Eq. 6.1]	$(\alpha, \beta, \gamma, 0)$
Algorithm in (6.3)	$(\alpha, \beta, \gamma, \delta)$

Lemma 19. *For $f \in \mathcal{S}_{m,L}$ with minimizer $x_\star \in \mathbb{R}^n$, we have $f(x) - f(x_\star) \leq \frac{L}{2}\|x - x_\star\|^2$ for all $x \in \mathbb{R}^n$.*

Proof. Since ∇f is Lipschitz continuous with Lipschitz constant L,

$$|f(x) - f(y) - (\nabla f(y))^T(y - x)| \leq \frac{L}{2}\|x - y\|^2$$

for all $x, y \in \mathbb{R}^n$ [**48**, Lemma 1.2.3]. The result follows from substituting $y = x_\star$. □

The following theorem characterizes the convergence of both the iterates and the corresponding function values of gradient descent. We provide the proof in Section 6.4. While the result for gradient descent is not new, we provide the proof as a comparison for our proposed algorithm.

Theorem 20 (Gradient descent)**.** *For all $k \geq 0$, let $f_k \in \mathcal{S}_{m,L}$ with $0 < m \leq L$ where $0 = \nabla f_k(x_\star)$. For any initial condition $\xi_0 \in \mathbb{R}^n$, the gradient descent method produces iterates which satisfy*

$$\|x_k - x_\star\| \leq \rho^k \|x_0 - x_\star\| \tag{6.4}$$

$$f_k(x_k) - f_k(x_\star) \leq \rho^{2k} \frac{L}{2} \|x_0 - x_\star\|^2 \tag{6.5}$$

for all $k \geq 0$ where

$$\rho = \begin{cases} 1 - \dfrac{1}{\kappa}, & \text{if } \alpha = \dfrac{1}{L} \\ \dfrac{\kappa - 1}{\kappa + 1}, & \text{if } \alpha = \dfrac{2}{L+m}. \end{cases} \tag{6.6}$$

Our proposed algorithm, called the *triple momentum method*, generalizes gradient descent using the form in (6.3) with specific choices for the parameters.

Definition 22 (Triple momentum method)**.** *Let $\rho = 1 - 1/\sqrt{\kappa}$. We call the algorithm in* (6.3) *with constant parameters*

$$(\alpha, \beta, \gamma, \delta) = \left(\frac{1+\rho}{L}, \ \frac{\rho^2}{2-\rho}, \ \frac{\rho^2}{(1+\rho)(2-\rho)}, \ \frac{\rho^2}{1-\rho^2} \right) \tag{6.7}$$

the triple momentum method *(or TM method).*

We now state our main theorem which gives bounds on the error of both the iterates and the function values for the TM method. The proof is in Section 6.7.

Theorem 21 (Triple momentum method)**.** *Let $f \in \mathcal{S}_{m,L}$ with $0 < m \leq L$. Let $x_\star \in \mathbb{R}^n$ be the unique minimizer of f. For any initial condition $\xi_0, \xi_{-1} \in \mathbb{R}^n$, the TM method produces iterates which satisfy*

$$\|x_k - x_\star\| \leq \rho^{k-1} \|x_1 - x_\star\| \tag{6.8}$$

$$f(x_k) - f(x_\star) \leq \rho^{2(k-1)} \frac{L}{2} \|x_1 - x_\star\|^2 \tag{6.9}$$

for all $k \geq 1$.

Using integral quadratic constraints, we can also prove the following bounds for the TM method. The proof of Theorem 22 is given in Section 6.5.

Theorem 22 (Triple momentum method)**.** *Let $f \in \mathcal{S}_{m,L}$ with $0 < m \le L$. Let $x_\star \in \mathbb{R}^n$ be the unique minimizer of f. For any initial condition $\xi_0, \xi_{-1} \in \mathbb{R}^n$, the TM method produces iterates which satisfy*

$$\|x_k - x_\star\| \le \rho^k \sqrt{\|x_0 - x_\star\|^2 + \frac{c}{m}} \tag{6.10}$$

$$f(x_k) - f(x_\star) \le \rho^{2k} \frac{L}{2} \left(\|x_0 - x_\star\|^2 + \frac{c}{m} \right) \tag{6.11}$$

for all $k \ge 1$ where

$$c = \left[\nabla f(y_0) - m(y_0 - x_\star)\right]^T \left[\frac{\gamma}{\beta} \xi_0 + \left(1 - \frac{\gamma}{\beta}\right) \xi_{-1} - x_\star \right]. \tag{6.12}$$

Furthermore, if $\xi_0 = \xi_{-1}$, then the iterates satisfy

$$\|x_k - x_\star\| \le \rho^k \sqrt{\kappa}\, \|x_0 - x_\star\| \tag{6.13}$$

$$f(x_k) - f(x_\star) \le \rho^{2k} \frac{L}{2} \kappa\, \|x_0 - x_\star\|^2 \tag{6.14}$$

for all $k \ge 0$.

A few remarks concerning Theorem 22:

Remark 7. *The state ξ also converges to $x_\star$. The transfer function from ξ to x is $(1+\delta) - \delta\, z^{-1}$. The inverse of this transfer function has a pole at $\delta/(1+\delta)$. For any m and L with $0 < m \le L$, the pole is in the interval $[0, 1/3]$ so the transfer function is stable. Since* *the transfer function from x to ξ is stable with unit dc gain and x converges to $x_\star$, then ξ also converges to $x_\star$.*

Remark 8. *Simulations (not shown) indicate the algorithm is robust to the parameters in (6.7). If $f \in \mathcal{S}_{\tilde{m},\tilde{L}}$ (instead of $\mathcal{S}_{m,L}$), then the algorithm still converges linearly so long as $\tilde{m} > 0$ and $\tilde{L}$ is not too large, although the convergence may be slower.*

Remark 9. *The value of c in (6.12) may be negative, however, the quanitity $\|x_0 - x_\star\|^2 + c/m$ is always nonnegative. For this reason, the bounds in (6.10) and (6.11) may not hold for $k = 0$ (but do hold for $k \geq 1$).*

For comparison, we plot the convergence rates of the iterates and the number of iterations required to obtain a given tolerance for different methods in Fig. 6.1. To obtain an accuracy in the iterates of ϵ, we have $\|x_k - x_\star\| = \epsilon \leq c\rho^k$. The iterations required to converge is then

$$k \geq -\frac{\ln(c/\epsilon)}{\ln \rho} \propto -\frac{1}{\ln \rho}. \tag{6.15}$$

For ill-conditioned problems, the condition ratio is large. In this case, the convergence rate is approximately one, so we can use the approximation $\ln(1+x) \approx x$ for small x to obtain

$$k \propto \frac{1}{1-\rho}, \qquad \kappa \text{ large.} \tag{6.16}$$

This approximation yields the approximate iterations to converge for ill-conditioned problems in Table 6.2.

We now develop the integral quadratic constraint tools which will be used to prove Theorems 20 and 22. These allow us to characterize ∇f when $f \in \mathcal{S}_{m,L}$ and can be used to both analyze and design algorithms of the form (6.3).

Table 6.2. Approximate iterations to converge for gradient optimization algorithms for large κ (ignoring constant terms).

Method	Iterations to obtain $\|x_k - x_\star\| \leq \epsilon$
Gradient descent, $\alpha = 1/L$	κ
Gradient descent, $\alpha = 2/(L+m)$	$\kappa/2$
Nesterov's method	$2\sqrt{\kappa}$
Triple momentum method	$\sqrt{\kappa}$
Theoretical lower bound	$\sqrt{\kappa}/2$

6.2. Analysis using Integral Quadratic Constraints

Integral quadratic constraints (IQCs) are a powerful tool for analyzing interconnected dynamical systems which contain nonlinear components, including gradient-based optimization algorithms [**49**] and estimators for distributed average tracking. The basic setup for such systems is shown in Fig. 6.2 where G is a linear system and Δ is either an unknown or nonlinear function which is difficult to analyze. For example, Δ is the gradient of the objective function in gradient-based optimization algorithms and the nonlinear Laplacian operator in dynamic average consensus estimators.

Consider an unknown function $\Delta : \mathbb{R}^n \to \mathbb{R}^n$ in feedback with a known linear system $G : \mathbb{R}^d \times \mathbb{R}^n \to \mathbb{R}^d \times \mathbb{R}^n$, where G is given by the recursion

$$\begin{aligned} \eta_{k+1} &= A\eta_k + B_1 u_k + B_2 v_k, \quad \eta_0 \in \mathbb{R}^p \\ y_k &= C_1 \eta_k + D_1 u_k \\ w_k &= C_2 \eta_k + D_2 u_k. \end{aligned} \tag{6.17}$$

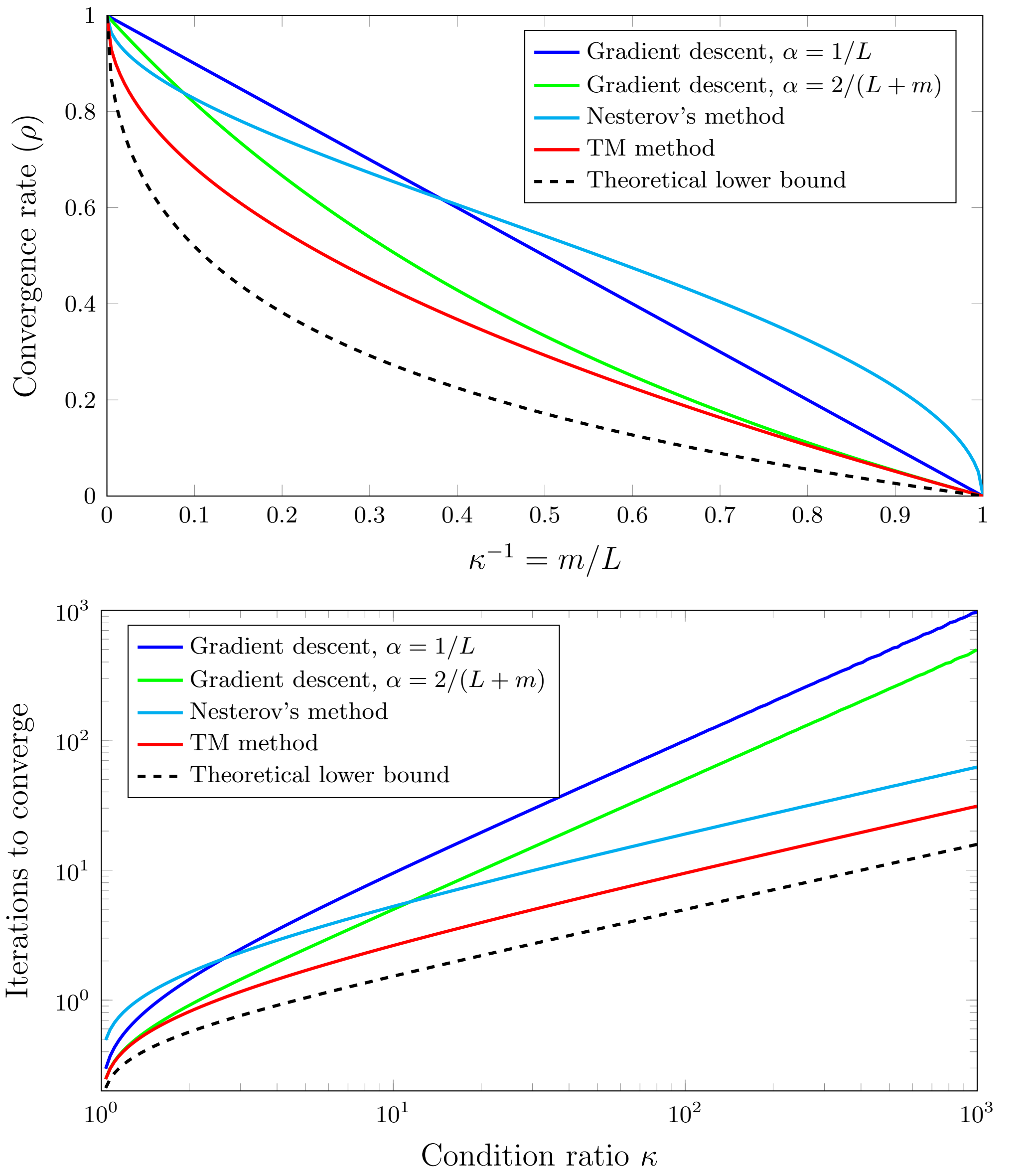


Figure 6.1. Theoretical properties of gradient optimization algorithms for $f \in \mathcal{S}_{m,L}$. Shown are gradient descent with $\alpha = 1/L$ (blue) and $\alpha = 2/(L+m)$ (green), Nesterov's method with $\alpha = 1/L$ and $\beta = (\sqrt{L} - \sqrt{m})/(\sqrt{L} + \sqrt{m})$ (cyan), and the TM method (red). Nesterov's lower bound (dashed black) is also shown. The heavy-ball method with $\alpha = 4/(\sqrt{L} + \sqrt{m})^2$ and $\beta = (\sqrt{L} - \sqrt{m})/(\sqrt{L} + \sqrt{m})$ converges *locally* with rate equal to the lower bound, but does not converge *globally*.

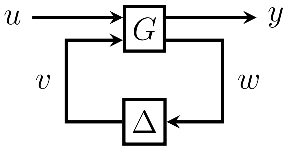


Figure 6.2. Block diagram of a known linear system G in feedback with an unknown function Δ. The system has input u and output y.

The internal state is $\eta \in \ell_{2e}^p$, the inputs are $u \in \ell_{2e}^d$ and $v \in \ell_{2e}^n$, and the outputs are $y \in \ell_{2e}^d$ and $w \in \ell_{2e}^n$. The feedback is given by $v = \Delta(w)$ which produces the closed-loop system

$$\begin{aligned} \eta_{k+1} &= A\eta_k + B_1 u_k + B_2 \Delta(C_2 \eta_k + D_2 u_k), \quad \eta_0 \in \mathbb{R}^p \\ y_k &= C_1 \eta_k + D_1 u_k. \end{aligned} \tag{6.18}$$

We suppose the closed-loop system has a fixed point $(\eta_\star, u_\star, v_\star, y_\star, w_\star)$, i.e.,

$$\begin{aligned} \eta_\star &= A\eta_\star + B_1 u_\star + B_2 v_\star \\ y_\star &= C_1 \eta_\star + D_1 u_\star \\ w_\star &= C_2 \eta_\star + D_2 u_\star \\ v_\star &= \Delta(w_\star), \end{aligned} \tag{6.19}$$

and we want a method for proving that the algorithm converges to the fixed point.

Example 3 (Gradient descent)**.** *Gradient descent is equivalent to* (6.18) *with* $\eta_0 = \xi_0$*,* $\Delta = (\nabla f_0, \nabla f_1, \ldots)$*,* $p = n$*, and*

$$G = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_1 & 0 \\ C_2 & D_2 & 0 \end{array}\right] = \left[\begin{array}{c|cc} 1 & 0 & -\alpha \\ \hline 1 & 0 & 0 \\ 1 & 0 & 0 \end{array}\right] \otimes I_n. \tag{6.20}$$

Example 4 (TM method)**.** *The TM method is equivalent to* (6.18) *with* $\eta_0 = \begin{bmatrix}\xi_0^T, \ \xi_{-1}^T\end{bmatrix}^T$*,* $\Delta = (\nabla f_0, \nabla f_1, \ldots)$*,* $p = 2n$*, and*

$$G = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_1 & 0 \\ C_2 & D_2 & 0 \end{array}\right] = \left[\begin{array}{cc|cc} 1+\beta & -\beta & 0 & -\alpha \\ 1 & 0 & 0 & 0 \\ \hline 1+\delta & -\delta & 0 & 0 \\ 1+\gamma & -\gamma & 0 & 0 \end{array}\right] \otimes I_n. \tag{6.21}$$

The system in (6.18) is difficult to analyze due to the unknown function Δ. The idea behind IQCs is to replace Δ with constraints that we know its input and output sequences must satisfy. If a result holds for any signals (w, v) which satisfy the constraints, then the result must also hold for the original system. To develop the constraints on (w, v), consider a linear system $\Psi : \ell_{2e}^n \times \ell_{2e}^n \to \ell_{2e}^m$ of the form

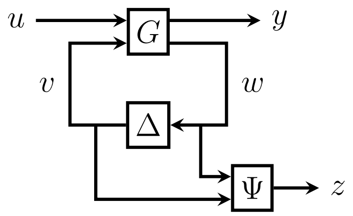
$$\begin{aligned} \zeta_{k+1} &= A_\Psi \zeta_k + B_\Psi^w w_k + B_\Psi^v v_k, \quad \zeta_0 = \zeta_\star \in \mathbb{R}^q \\ z_k &= C_\Psi \zeta_k + D_\Psi^w w_k + D_\Psi^v v_k \end{aligned} \tag{6.22}$$

where $\rho(A_\psi) < 1$ and $(\zeta_\star, z_\star)$ is the unique solution to

$$\begin{aligned} \zeta_\star &= A_\Psi \zeta_\star + B_\Psi^w w_\star + B_\Psi^v v_\star \\ z_\star &= C_\Psi \zeta_\star + D_\Psi^w w_\star + D_\Psi^v v_\star. \end{aligned} \tag{6.23}$$

This defines the map $z = \Psi(w, v)$ as shown in Fig. 6.3.

We now define a ρ-IQC, which is a constraint on z which will be used to prove linear convergence with rate ρ.



$$\begin{aligned} u, y &\in \ell_{2e}^d \\ v, w &\in \ell_{2e}^n \\ z &\in \ell_{2e}^m \\ \eta &\in \ell_{2e}^p \\ \zeta &\in \ell_{2e}^q \end{aligned}$$

Figure 6.3. Block diagram showing the IQC setup. The linear system G with state η is in feedback with the unknown function Δ. The system has input u and output y. The auxiliary system Ψ with state ζ filters w and v to produce an output z which satisfies the IQC. For reference, the space of each signal is given.

Definition 23 (ρ-IQC, [**49**, Defn. 3])**.** *Suppose $\rho \in [0,1]$, $\Delta : \ell_{2e}^n \to \ell_{2e}^n$ is an unknown map, and $v_\star, w_\star \in \mathbb{R}^n$ with $v_\star = \Delta(w_\star)$. Let Ψ and $(\zeta_\star, z_\star)$ be given by Eqs.* (6.22) *and* (6.23)*, respectively. Suppose $w \in \ell_2^n$ is an arbitrary square-summable sequence, i.e., $\sum_{k=0}^\infty \|w_k\|^2 < \infty$. Let $v = \Delta(w)$ and $z = \Psi(w,v)$. We say that $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$ if*

$$0 \le \sum_{j=0}^{k} \rho^{-2j} (z_j - z_\star)^T M (z_j - z_\star), \quad \textit{for all } k \ge 0. \tag{6.24}$$

Remark 10. *If $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$, then $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \tilde\rho)$ for any $\tilde\rho \ge \rho$.*

If $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$, then we can remove Δ from the block diagram and simply study the connection of the linear systems G and Ψ subject to the IQC constraint in (6.24). Therefore, we define the auxiliary system $\hat G : \ell_{2e}^n \to \ell_{2e}^d \times \ell_{2e}^m$ given by

$$\begin{aligned} \hat\xi_{k+1} &= \hat A \hat\xi_k + \hat B_1 u_k + \hat B_2 v_k \\ y_k &= \hat C_1 \hat\xi_k + \hat D_{11} u_k \\ z_k &= \hat C_2 \hat\xi_k + \hat D_{21} u_k + \hat D_{22} v_k \end{aligned} \tag{6.25}$$

where $\hat{\xi} = [\eta^T,\ \zeta^T]^T$ and

$$\left[\begin{array}{c|cc} \hat{A} & \hat{B}_1 & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{D}_{11} & 0 \\ \hat{C}_2 & \hat{D}_{21} & \hat{D}_{22} \end{array}\right] = \left[\begin{array}{cc|cc} A & 0 & B_1 & B_2 \\ B_\Psi^w C_2 & A_\Psi & B_\Psi^w D_2 & B_\Psi^v \\ \hline C_1 & 0 & D_1 & 0 \\ D_\Psi^w C_2 & C_\Psi & D_\Psi^w D_2 & D_\Psi^v \end{array}\right] \tag{6.26}$$

as shown in Fig. 6.4. The system is now completely characterized by the linear system $\hat{G}$ where $(y, z) = \hat{G}(u, v)$ along with the IQC constraint on z in (6.24).

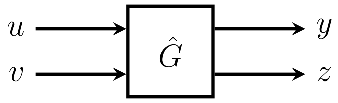


Figure 6.4. Equivalent system with the unknown function Δ replaced by an integral quadratic constraint on z. The system is then completely characterized by the linear system $\hat{G}$ and the IQC.

The system $\hat{G}$ may contain states which do not affect the output, i.e., it may be unobservable. In such cases, instead of using $\hat{G}$ to characterize the system, we can use any system $\bar{G}$ which produces the same output sequence as $\hat{G}$ for some initial condition. We also consider the case when some states of $\hat{G}$ do not affect the output after some finite number of iterations. Specifically, let $\bar{G} : \ell_{2n}^d \times \ell_{2e}^n \to \ell_{2e}^d \times \ell_{2e}^m$ be any system of the form

$$\begin{aligned} \bar{\xi}_{k+1} &= \bar{A}\bar{\xi}_k + \bar{B}_1 u_k + \bar{B}_2 v_k \\ \bar{y}_k &= \bar{C}_1 \bar{\xi}_k + \bar{D}_{11} u_k \\ \bar{z}_k &= \bar{C}_2 \bar{\xi}_k + \bar{D}_{21} u_k + \bar{D}_{22} v_k \end{aligned} \tag{6.27}$$

such that, for some initial condition $\bar{\xi}_0$ and some $k_0 \geq 0$,

$$\begin{bmatrix} \bar{y}_k \\ \bar{z}_k \end{bmatrix} = \begin{bmatrix} y_k \\ z_k \end{bmatrix} \quad \text{for all } k \geq k_0 \geq 0. \tag{6.28}$$

Note that the trivial choice $\bar{G} = \hat{G}$ satisfies these conditions with $k_0 = 0$ and $\bar{\xi}_0 = \hat{\xi}_0$.

We are now ready to present the main result of this section which gives conditions for global linear convergence of the output x to a fixed point of the system with rate ρ.

Theorem 23 (Main IQC result). *Consider the system* (6.18). *Suppose u is constant, i.e., $u_k = u$ for all $k \geq 0$, and suppose $(\eta_\star, u_\star, v_\star, y_\star, w_\star)$ is a fixed point of the system, i.e., satisfies* (6.19). *Let $\bar{G}$ be any system of the form in* (6.27) *such that the condition in* (6.28) *is satisfied for some initial condition $\bar{\xi}_0$ and some integer $k_0 \geq 0$ where y_k and z_k are defined by* (6.25). *If $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$ and there exists $P \succ 0$ such that*

$$0 \succeq \begin{bmatrix} \bar{C}_1\bar{A} & \bar{C}_1\bar{B}_2 \\ \rho\bar{C}_1 & 0 \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & -P \end{bmatrix} \begin{bmatrix} \bar{C}_1\bar{A} & \bar{C}_1\bar{B}_2 \\ \rho\bar{C}_1 & 0 \end{bmatrix} + \begin{bmatrix} \bar{C}_2 & \bar{D}_{22} \end{bmatrix}^T M \begin{bmatrix} \bar{C}_2 & \bar{D}_{22} \end{bmatrix}, \tag{6.29}$$

then for any initial condition $\eta_0 \in \mathbb{R}^p$, we have

$$\|x_k - x_\star\| \leq \rho^k \sqrt{\mathrm{cond}(P)\|\bar{x}_0 - x_\star\|^2 + \frac{\nu}{\sigma_{min}(P)}} \tag{6.30}$$

for all $k \geq k_0$ where $\bar{x}_0 = \bar{C}_1\bar{\xi}_0$ and

$$\nu := \sum_{j=0}^{k_0-1} \rho^{-2(j+1)} \Big[(z_j - z_\star)^T M (z_j - z_\star) - (\bar{z}_j - z_\star)^T M (\bar{z}_j - z_\star)\Big]. \tag{6.31}$$

Proof. Let $u, \bar{x}, \bar{z}, \bar{\xi} \in \ell_{2e}$ satisfy (6.27) and (6.28) with the initial condition $\bar{\xi}_0$. Suppose $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$ and $P \succ 0$ is a solution of (6.29). From Eqs. (6.19) and (6.28), there exists a vector $\bar{\xi}_\star$ which satisfies

$$\begin{aligned}\bar{\xi}_\star &= \bar{A}\bar{\xi}_\star + \bar{B}_1 u_\star + \bar{B}_2 v_\star \\ y_\star &= \bar{C}_1\bar{\xi}_\star + \bar{D}_{11} u_\star \\ z_\star &= \bar{C}_2\bar{\xi}_\star + \bar{D}_{21} u_\star + \bar{D}_{22} v_\star .\end{aligned} \tag{6.32}$$

Then define the quantity

$$\mu_k := (\bar{y}_{k+1} - y_\star)^T P (\bar{y}_{k+1} - y_\star) - \rho^2 (\bar{y}_k - y_\star)^T P (\bar{y}_k - y_\star) + (\bar{z}_k - z_\star)^T M (\bar{z}_k - z_\star). \tag{6.33}$$

From (6.27) and (6.32), we have

$$\begin{aligned}\mu_k = &\begin{bmatrix} \star \\ \star \\ \star \end{bmatrix}^T \begin{bmatrix} \star & \star & \star \\ \star & \star & \star \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & -P \end{bmatrix} \begin{bmatrix} \bar{C}_1\bar{A} & \bar{C}_1\bar{B}_1 + \bar{D}_{11} & \bar{C}_1\bar{B}_2 \\ \rho\bar{C}_1 & \rho\bar{D}_{11} & 0 \end{bmatrix} \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ u_k - u_\star \\ v_k - v_\star \end{bmatrix} \\ &+ \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ u_k - u_\star \\ v_k - v_\star \end{bmatrix}^T \begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & \bar{D}_{22} \end{bmatrix}^T M \begin{bmatrix} \bar{C}_2 & \bar{D}_{21} & \bar{D}_{22} \end{bmatrix} \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ u_k - u_\star \\ v_k - v_\star \end{bmatrix}\end{aligned} \tag{6.34}$$

where $\star$ denotes the repeated part of the quadratic form. Since the input is constant, $u_k = u_\star$ for all $k \geq 0$, so

$$\begin{aligned}\mu_k = &\begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ v_k - v_\star \end{bmatrix}^T \begin{bmatrix} \bar{C}_1\bar{A} & \bar{C}_1\bar{B}_2 \\ \rho\bar{C}_1 & 0 \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & -P \end{bmatrix} \begin{bmatrix} \bar{C}_1\bar{A} & \bar{C}_1\bar{B}_2 \\ \rho\bar{C}_1 & 0 \end{bmatrix} \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ v_k - v_\star \end{bmatrix} \\ &+ \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ v_k - v_\star \end{bmatrix}^T \begin{bmatrix} \bar{C}_2 & \bar{D}_{22} \end{bmatrix}^T M \begin{bmatrix} \bar{C}_2 & \bar{D}_{22} \end{bmatrix} \begin{bmatrix} \bar{\xi}_k - \bar{\xi}_\star \\ v_k - v_\star \end{bmatrix} .\end{aligned} \tag{6.35}$$

The matrix in (6.29) is negative semidefinite, so $0 \geq \mu_k$ for all $k \geq 0$. Multiply μ_j by $\rho^{2(k-j-1)}$ and sum over $j = 0, \dots, k-1$. The first two terms yield a telescoping sum, and we obtain

$$0 \geq (\bar{x}_k - x_\star)^T P(\bar{x}_k - x_\star) - \rho^{2k}(\bar{x}_0 - x_\star)^T P(\bar{x}_0 - x_\star) + \rho^{2(k-1)} \sum_{j=0}^{k-1} \rho^{-2j} (\bar{z}_j - z_\star)^T M(\bar{z}_j - z_\star).$$

Define $\tilde{x}_k := x_k - x_\star$ and $\tilde{z}_k := z_k - z_\star$. Then using the conditions in (6.28), we have

$$\tilde{x}_k^T P \tilde{x}_k \leq \rho^{2k}(\bar{x}_0 - x_\star)^T P(\bar{x}_0 - x_\star) - \rho^{2(k-1)} \sum_{j=0}^{k-1} \rho^{-2j} \tilde{z}_j^T M \tilde{z}_j + \rho^{2k}\nu \tag{6.36}$$

for $k \geq k_0$ where ν is defined in (6.31). Because $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$, the summation term in (6.36) is nonnegative for all $k \geq 0$, so

$$(x_k - x_\star)^T P(x_k - x_\star) \leq \rho^{2k}\big[(\bar{x}_0 - x_\star)^T P(\bar{x}_0 - x_\star) + \nu\big].$$

Multiplying by $\|x_k - x_\star\|^2/(x_k - x_\star)^T P(x_k - x_\star)$ gives

$$\|x_k - x_\star\|^2 \leq \rho^{2k}\left[\mathrm{cond}(P)\|\bar{x}_0 - x_\star\|^2 + \frac{\nu}{\sigma_{\min}(P)}\right].$$

Taking the square root gives the bound in (6.30). □

Theorem 23 extends [**49**, Thm. 4] in the following ways:

(1) The system has an input u_k (which is assumed to be constant in Thm. 23).

(2) The output is a function of the state of G given by $y_k = C_1\xi_k + D_1u_k$. Then $P \in \mathbb{S}^d$ since P multiplies x, whereas $P \in \mathbb{S}^{p+q}$ in [**49**] since it multiplies the combined state $[\eta^T \ \zeta^T]^T$. The smaller P matrix makes it easier to obtain closed-form solutions. Also, C_1 can be chosen to minimize cond(P) to improve the bound in (6.30).

(3) Instead of $\hat{G}$, an equivalent (possibly lower order) system $\bar{G}$ can be used.

Remark 11. *It is sometimes possible to obtain a tighter bound on the error by applying Theorem 23 to a set of values for ρ and taking the minimum over each bound.*

By applying Theorem 23 with $\bar{G} = \hat{G}$, we obtain the simple corollary.

Corollary 8. *Consider the system* (6.18). *Suppose u is constant, i.e., $u_k = u$ for all $k \geq 0$, and suppose $(\eta_\star, u_\star, v_\star, y_\star, w_\star)$ is a fixed point of the system, i.e., satisfies* (6.19). *If $\Delta \in \mathrm{IQC}(\Psi, M, v_\star, w_\star, \rho)$ and there exists $P \succ 0$ such that*

$$0 \succeq \begin{bmatrix} \hat{C}_1\hat{A} & \hat{C}_1\hat{B}_2 \\ \rho\hat{C}_1 & 0 \end{bmatrix}^T \begin{bmatrix} P & 0 \\ 0 & -P \end{bmatrix} \begin{bmatrix} \hat{C}_1\hat{A} & \hat{C}_1\hat{B}_2 \\ \rho\hat{C}_1 & 0 \end{bmatrix} + \begin{bmatrix} \hat{C}_2 & \hat{D}_{22} \end{bmatrix}^T M \begin{bmatrix} \hat{C}_2 & \hat{D}_{22} \end{bmatrix}, \tag{6.37}$$

then for any initial condition $\eta_0 \in \mathbb{R}^p$, we have

$$\|x_k - x_\star\| \leq \rho^k \sqrt{\mathrm{cond}(P)}\, \|x_0 - x_\star\| \tag{6.38}$$

for all $k \geq 0$.

Next, we provide several useful IQCs which characterize ∇f when $f \in \mathcal{S}_{m,L}$.

Lemma 20 ([**49**, Lemma 6])**.** *Suppose $f_k \in \mathcal{S}_{m,L}$ for each k and $(u_\star, y_\star)$ is a common reference point for the gradients of f_k, i.e., $u_\star = \nabla f_k(y_\star)$ for all $k \geq 0$. Let $\Delta = (\nabla f_0, \nabla f_1, \ldots)$. Then $\Delta \in \mathrm{IQC}(\Psi, M, u_\star, y_\star, 0)$ where*

$$\Psi = \begin{bmatrix} L & -1 \\ -m & 1 \end{bmatrix} \otimes I_n, \qquad M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes I_n. \tag{6.39}$$

When f is constant, the IQC can be generalized by introducing dynamics in Ψ.

Lemma 21 ([**49**, Lemma 10])**.** *Suppose $f \in \mathcal{S}_{m,L}$ and $(u_\star, y_\star)$ is a reference point for the gradient of f, i.e., $u_\star = \nabla f(y_\star)$. Let $H(z) = \bar{\rho}^2/z$ with $\bar{\rho} \in [0,1]$. Let $\Delta = (\nabla f, \nabla f, \ldots)$. Then $\Delta \in \mathrm{IQC}(\Psi, M, u_\star, y_\star, \rho)$ for any $\rho \in [\bar{\rho}, 1]$ where*

$$\Psi = \begin{bmatrix} L(1-H) & -(1-H) \\ -m & 1 \end{bmatrix} \otimes I_n, \qquad M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \otimes I_n. \tag{6.40}$$

Remark 12. *For $H(z) = \bar{\rho}^2/z$, a parameterization of Ψ in* (6.40) *is*

$$\left[\begin{array}{c|cc} A_\Psi & B_\Psi^y & B_\Psi^u \\ \hline C_\Psi & D_\Psi^y & D_\Psi^u \end{array}\right] = \left[\begin{array}{c|cc} 0 & -L & 1 \\ \hline \bar{\rho}^2 & L & -1 \\ 0 & -m & 1 \end{array}\right] \otimes I_n. \tag{6.41}$$

6.3. Design using Integral Quadratic Constraints

We now consider how to design systems using integral quadratic constraints of the form in (6.40). (This includes the IQC in Lemma 20 by setting $H(z) = 0$.) Conditions in terms of the system G, the IQC parameter H, the matrix parameter P, and the convergence rate ρ to guarantee global linear convergence of the algorithm with rate ρ are given in Theorem 23. The matrix inequality in (6.29) is not linear in all of the unknowns, however, and does not directly lead to a simple design procedure. In other words, Theorem 23 is good for analysis, but not for design. The difficulty with using Theorem 23 to design algorithms is in finding conditions under which (6.29) has a solution $P \succ 0$. Instead of using (6.29) directly, we design the algorithm using the following *relaxed* frequency domain condition. (We use the notation $\angle(re^{j\theta}) = \theta$ to denote the angle of a complex number.)

Lemma 22. *Suppose Ψ and M are given by* (6.40) *with $H(z) = \bar{\rho}^2/z$. Let $G_2(z) = C_2(zI - A)^{-1}B_2 = g_2(z) \otimes I_n$. Also, suppose the system*

$$\hat{G}_2(z) := \Psi(z) \begin{bmatrix} G_2(z) \\ I \end{bmatrix} = \left[\begin{array}{c|c} \hat{A} & \hat{B}_2 \\ \hline \hat{C}_2 & \hat{D}_2 \end{array}\right] \tag{6.42}$$

has no poles on $\rho\mathbb{T}$ and the pair $(\hat{A}, \hat{B}_2)$ is controllable. If

$$\angle F(z) \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \quad \textit{for all } z \in \rho\mathbb{T} \tag{6.43}$$

where

$$F(z) := \frac{1 - Lg_2(z)}{1 - mg_2(z)}(1 - H(z)), \tag{6.44}$$

then there exists $\tilde{P} \in \mathbb{S}^p$ such that the LMI in (6.29) *is satisfied with $\bar{G} = \hat{G}$ for any P and C_1 such that $\tilde{P} = \hat{C}_1^T P \hat{C}_1$.*

Proof. Suppose (6.43) is satisfied. For Ψ and M given by (6.40), we have

$$\hat{G}_2(z)^* M \hat{G}_2(z) = -2 \operatorname{Re}\{f(z)\} \otimes I_n$$

where

$$f(z) = (1 - mg_2(z^*))(1 - Lg_2(z))(1 - H(z)). \tag{6.45}$$

Note that $\angle F(z) = \angle f(z)$ since $\angle(1 - mg_2(z^*)) = -\angle(1 - mg_2(z))$. From (6.43), we have $\mathrm{Re}\{F(z)\} \geq 0$ for all $z \in \rho\mathbb{T}$ which implies $\mathrm{Re}\{f(z)\} \geq 0$. Then

$$\hat{G}_2(z)^* M \hat{G}_2(z) \leq 0 \quad \text{for all } z \in \rho\mathbb{T}.$$

Since $\hat{G}_2(z)$ has no poles on $\rho\mathbb{T}$ and the pair $(\hat{A}, \hat{B})$ is controllable, we can apply the discrete-time KYP Lemma [**71**] to the previous frequency domain inequality. This implies that there exists $\tilde{P} \in \mathbb{S}^p$ such that

$$0 \succeq \begin{bmatrix} \hat{A} & \hat{B}_2 \\ \rho I & 0 \end{bmatrix}^T \begin{bmatrix} \tilde{P} & 0 \\ 0 & -\tilde{P} \end{bmatrix} \begin{bmatrix} \hat{A} & \hat{B}_2 \\ \rho I & 0 \end{bmatrix} + \begin{bmatrix} \hat{C}_2 & \hat{D}_2 \end{bmatrix}^T M \begin{bmatrix} \hat{C}_2 & \hat{D}_2 \end{bmatrix}.$$

Therefore, (6.29) is satisfied with $\bar{G} = \hat{G}$ for any P and C_1 such that $\tilde{P} = \hat{C}_1^T P \hat{C}_1$. □

Remark 13. *Lemma 22 only ensures that* (6.29) *is satisfied for some values of* P *and* $\hat{C}_1$*, but does not guarantee that* $P \succ 0$ *as required in Theorem 23 or that* $\hat{C}_1 = [C_1\ 0]$ *as in* (6.25)*. Conditions exist on* (Ψ, M) *which guarantee that* $P \succeq 0$ [**72**]*. However, we simply use* (6.43) *to design the algorithm, not to prove convergence. Once the algorithm is designed using this relaxed condition, we show in Sections 6.4 and 6.5 (for gradient descent and the TM method, respectively) that we do have* $P \succ 0$ *and* $\hat{C}_1 = [C_1\ 0]$ *as required.*

Remark 14. *Frequency domain conditions which do guarantee global asymptotic stability for systems with sector nonlinearities are given in* [**73, 74**].

Condition (6.43) can be analyzed graphically as follows:

(1) Draw the root locus of $-g_2(z)$.

(2) Draw a pole/zero plot of $F(z)$ as follows:

(a) The root locus poles at gain m are poles of $F(z)$.

(b) The root locus poles at gain L are zeros of $F(z)$.

(c) The poles and zeros of $1 - H(z)$ are poles and zeros, respectively, of $F(z)$.

(3) For all $z \in \rho\mathbb{T}$, calculate $\angle F(z)$ by summing the angles from z to the zeros and subtracting the angles from z to the poles.

This graphical procedure can be used to design $G_2(z)$, the IQC parameter $H(z)$, and the convergence rate ρ. The remaining parameters C_1 and P are then obtained using (6.29). The design process is summarized as follows:

(1) Use the graphical procedure to design $G_2(z)$, $H(z)$, and ρ in the frequency domain to satisfy the condition in (6.43).

(2) The LMI in (6.29) is now linear in $\tilde{P} = \hat{C}_1^T P \hat{C}_1$ and can be solved efficiently for $\tilde{P}$. From Lemma 22, we know that a solution $\tilde{P} \in \mathbb{S}^p$ exists.

(3) If we can write $\tilde{P} = \hat{C}_1^T P \hat{C}_1$ with $P \succ 0$ and $\hat{C}_1 = [C_1 \ 0]$ where $P \in \mathbb{S}^d$ and $C_1 \in \mathbb{R}^{d \times p}$, then we can apply Theorem 23 to guarantee global linear convergence with rate ρ with error bound (6.30).

6.4. Gradient Descent

We now design and analyze the gradient descent method using the IQC tools from Sections 6.2 and 6.3. It is well-known that the optimal step size for gradient descent is $\alpha = 2/(L + m)$ and the corresponding convergence rate is $\rho = (L - m)/(L + m)$ (for example, see [**49**, Section 4.4]). However, we go through the design process so that we can compare with the TM method.

6.4.1. Design

Example 3 gives the parameters for gradient descent in the form of (6.18). The transfer function of G from the second input to the second output is then $g_2(z) \otimes I_n$ where

$$g_2(z) = -\frac{\alpha}{z-1}. \tag{6.46}$$

To prove the bounds for gradient descent in Theorem 20 when the function is time-varying, we must use the static IQC in Lemma 20 which corresponds to $H(z) = 0$.

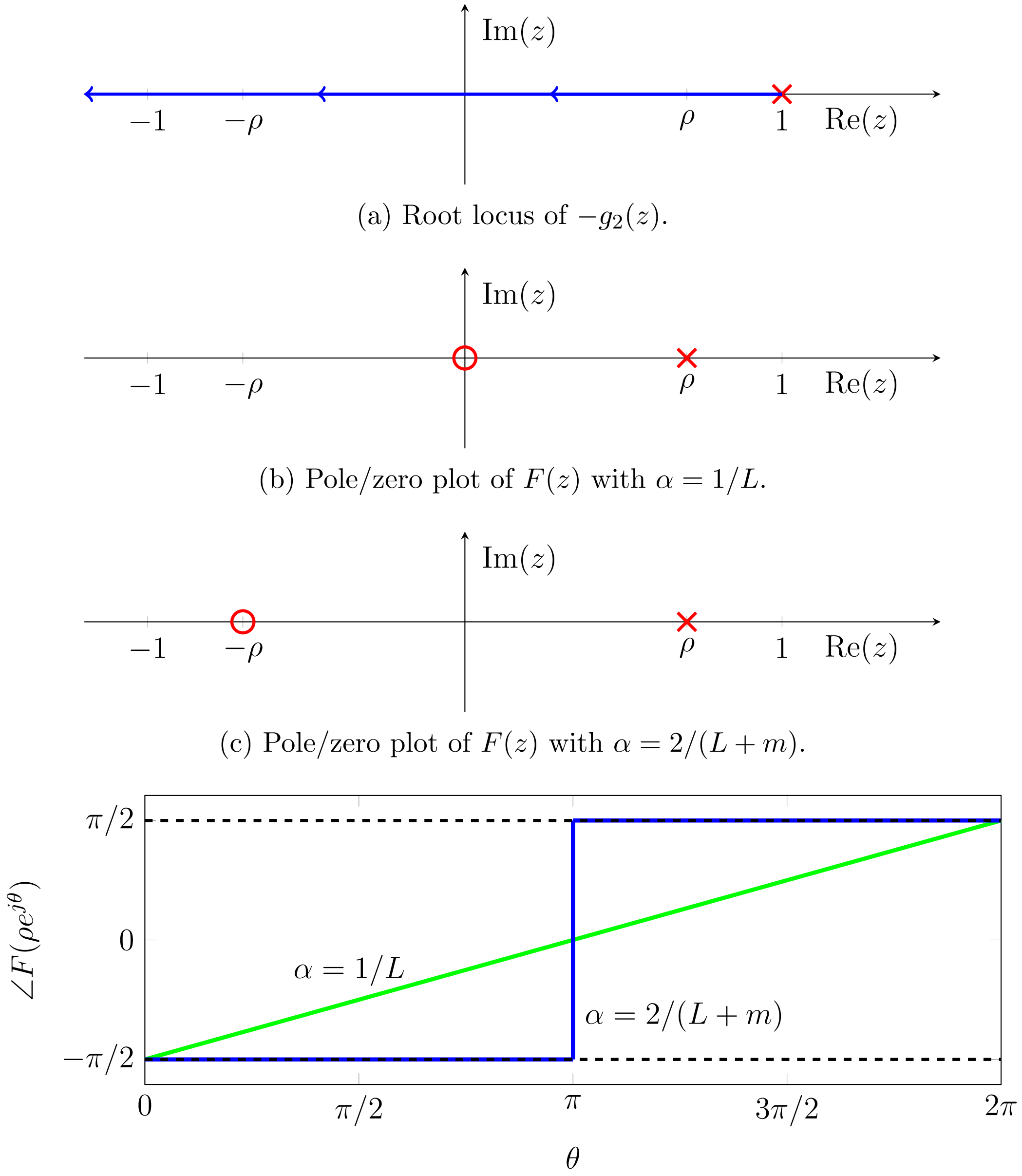
We now use the graphical design procedure to analyze the condition in (6.43) as follows. First, we draw the root locus of $-g_2(z)$ as shown in Fig. 6.5a. The root locus poles at gains m and L are the poles and zeros of $F(z)$, respectively. These are shown in Figs. 6.5b and 6.5c for two choices of the step size. Note that $H(z)$ does not contribute any phase since the IQC is static. We then sum the angle from z to the zero and subtract the angle from z to the pole for $z \in \rho\mathbb{T}$ to obtain the phase plot of $F(z)$ in Fig. 6.5d. From Lemma 22 we want the phase of $F(z)$ to be in the interval $[-\pi/2, \pi/2]$, which is satisfied for both choices of α.

From Fig. 6.5, it is clear that the angle condition is satisfied if and only if the pole and zero are both inside $\rho\mathbb{T}$. To obtain the smallest such ρ, we set the pole and zero on the boundary of $\rho\mathbb{T}$ using

$$0 = 1 - m g_2(\rho) \tag{6.47}$$

$$0 = 1 - L g_2(-\rho). \tag{6.48}$$

Solving these two equations gives the optimal step size $\alpha = 2/(L+m)$ and convergence rate $\rho = (L-m)/(L+m)$.



(a) Root locus of $-g_2(z)$.

(b) Pole/zero plot of $F(z)$ with $\alpha = 1/L$.

(c) Pole/zero plot of $F(z)$ with $\alpha = 2/(L+m)$.

(d) Phase plot of $F(\rho e^{j\theta})$ with $\alpha = 1/L$ (green) and $\alpha = 2/(L+m)$ (blue).

Figure 6.5. Design of gradient descent.

6.4.2. Analysis (Proof of Theorem 20)

We now prove the bounds in Theorem 20 using the IQC tools from Section 6.2. From Example 3, gradient descent has the form in (6.18) with $\Delta = \nabla f$, $\eta = \xi$, $p = n$, and G given by (6.20).

Since $f_k \in \mathcal{S}_{m,L}$ for all k, the static IQC in Lemma 20 gives that $\nabla f \in \text{IQC}(\Psi, M, u_\star, y_\star, \rho)$ where Ψ and M are give by (6.39) and $u_\star = \nabla f(y_\star)$. Since $f(x)$ is strongly convex, it has a unique minimizer $x_\star$. The combined system $\hat{G}$ in (6.26) is

$$\left[\begin{array}{c|cc} \hat{A} & \hat{B}_1 & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{D}_{11} & 0 \\ \hat{C}_2 & \hat{D}_{21} & \hat{D}_{22} \end{array}\right] = \left[\begin{array}{c|cc} 1 & 0 & -\alpha \\ \hline 1 & 0 & 0 \\ L & 0 & -1 \\ -m & 0 & 1 \end{array}\right] \otimes I_n \tag{6.49}$$

where the state is $\hat{\xi} = \eta = \xi$. In this case we can use Corollary 8 to prove the result. The matrix in (6.37) is then

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} P + \begin{bmatrix} -2Lm & L+m \\ L+m & -2 \end{bmatrix}. \tag{6.50}$$

If this matrix is negative semidefinite, then we can use Corollary 8 to prove that the algorithm converges to a fixed point which satisfies the IQC. The only point which is both a fixed point of the system and such that the gradient of f satisfies the IQC is

$$(u_\star, x_\star, y_\star, z_\star, \eta_\star, \zeta_\star) = \left(\mathbf{0}_n, x_\star, x_\star, \begin{bmatrix} Lx_\star \\ -mx_\star \end{bmatrix}, x_\star, \mathbf{0}_{0\times 0} \right). \tag{6.51}$$

In particular, the fixed point for x is the minimizer $x_\star$. Since P is a scalar in this case, it has unit condition number, i.e., $\text{cond}(P) = 1$. Equation (6.38) provides the bound on the iterates, and Lemma 19 gives the corresponding bound on the function values.

We have left to show that the matrix in (6.50) is negative semidefinite for both step sizes.

6.4.3. $\alpha = 1/L$

Using the parameters $\alpha = 1/L$, $\rho = 1 - m/L$, and $P = L^2$, the matrix in (6.50) is

$$\begin{bmatrix} 1-\rho^2 & -\alpha \\ -\alpha & \alpha^2 \end{bmatrix} P + \begin{bmatrix} -2Lm & L+m \\ L+m & -2 \end{bmatrix} = \begin{bmatrix} -m^2 & m \\ m & -1 \end{bmatrix} \tag{6.52}$$

which has eigenvalues at zero and $-(1+m^2)$, and is therefore negative semidefinite.

6.4.4. $\alpha = 2/(L+m)$

Using the parameters $\alpha = 2/(L+m)$, $\rho = (L-m)/(L+m)$, and $P = (L+m)^2/2$, the matrix in (6.50) is the zero matrix, and is therefore negative semidefinite.

6.5. Triple Momentum Method

We now design and analyze the TM method similar to gradient descent.

6.5.1. Design

From Example 4, the TM method is of the form in (6.18) with $G(z)$ given by (6.21). The transfer function of G from the second input to the second output is then

$$g_2(z) = -(1+\gamma)\alpha \frac{\left(z - \frac{\gamma}{1+\gamma}\right)}{(z-1)(z-\beta)}. \tag{6.53}$$

In this case, f is constant so we can use the dynamic IQC in Lemma 21 with $H(z) = \bar{\rho}^2/z$. To design the algorithm, we first consider how to design $H(z)$, i.e., how should we choose $\bar{\rho}$? Since we need ∇f to satisfy the ρ-IQC, we need $\bar{\rho} \le \rho$. From (6.44), the phase of $1 - H(z)$ is added to the phase of $(1 - Lg_2(z))/(1 - mg_2(z))$ to obtain the phase of $F(z)$. The phase of $1 - H(z)$ is plotted in Fig. 6.6 for several choices of $H(z)$. From the root locus of $-g_2(z)$ in

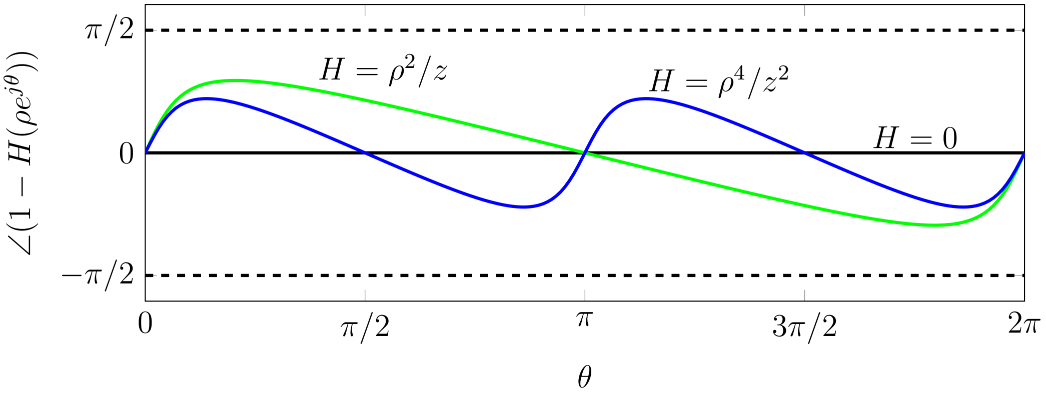


Figure 6.6. Phase plot of $\angle(1 - H(\rho e^{j\theta}))$ for $H(z) = 0$ (black), $H(z) = \rho^2/z$ (green), and $H(z) = \rho^4/z^2$ (blue) with $\rho = 0.8$.

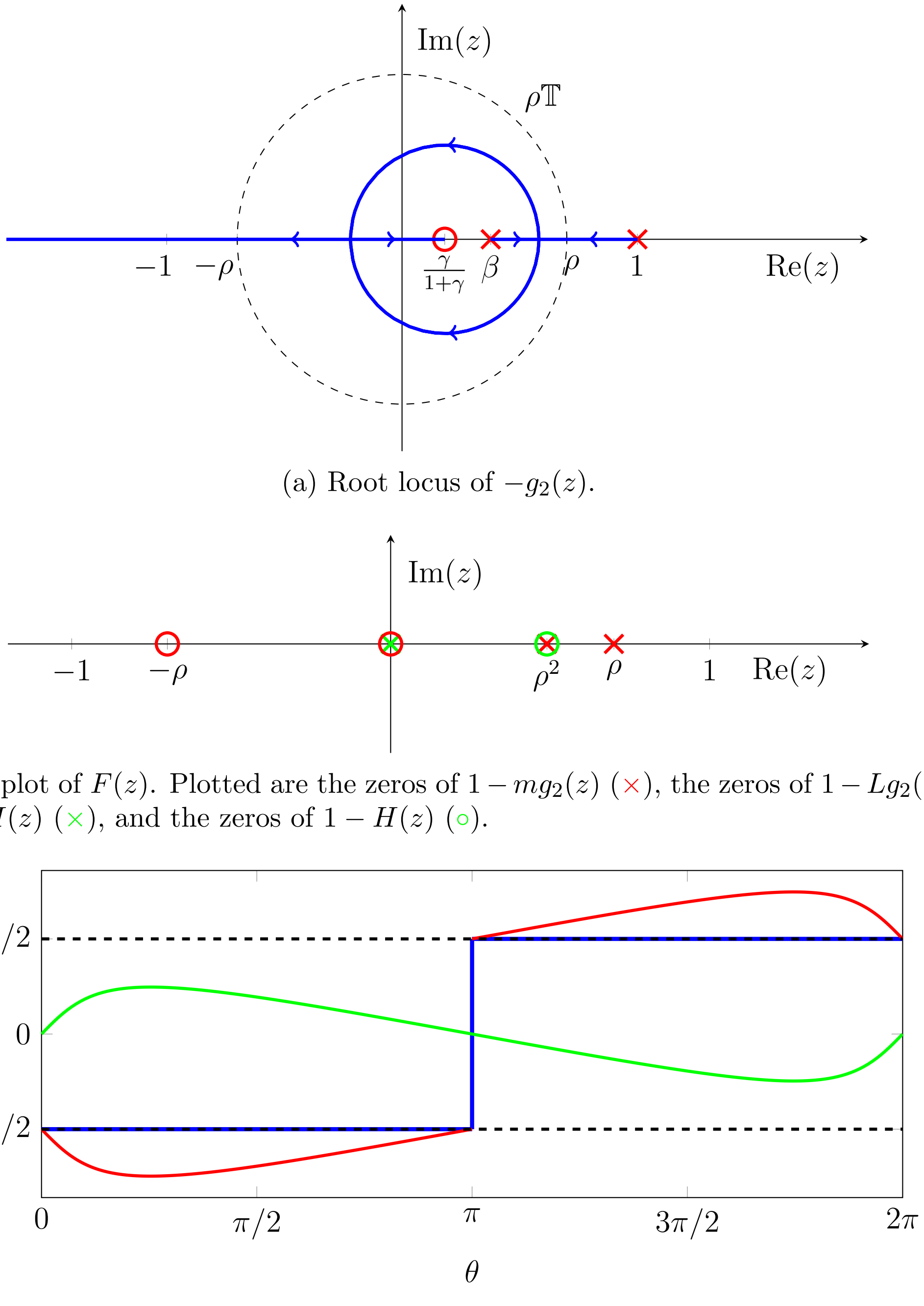
Fig. (6.7a), it is clear that $(1 - Lg_2(z))/(1 - mg_2(z))$ will have negative phase for $\theta \in (0, \pi)$ and positive phase for $\theta \in (\pi, 2\pi)$ (see Fig. 6.7c). Therefore, we want to choose $H(z)$ to have large positive phase in $(0, \pi)$ and large negative phase in $(\pi, 2\pi)$. The largest amount of phase is achieved using $H(z) = \rho^2/z$ (see Fig. 6.6).

We now consider how to design the parameters α, β, γ, and ρ. Since $H(z) = \rho^2/z$, then $1 - H(z)$ has a zero at $z = \rho^2$ and a pole at $z = 0$. We design $g_2(z)$ to cancel the pole and zero of $1 - H(z)$ and to have a pole and zero at ρ and $-\rho$, similar to gradient descent, as shown in Fig. 6.7a. In other words,

- Design $1 - mg_2(z)$ to have roots at $z = \rho^2$ (to cancel the zero of $1 - H(z)$) and $z = \rho$.
- Design $1 - Lg_2(z)$ to have roots at $z = 0$ (to cancel the pole of $1 - H(z)$) and $z = -\rho$.

These four conditions are used to solve for the four parameters α, β, γ, and ρ.

To obtain the other parameters δ and P, we solve (6.29) in closed-form by searching for parameters which make the matrix in (6.29) low rank. The TM method has enough degrees of freedom that the parameters can be chosen such that the matrix has rank zero, i.e., it is



(a) Root locus of $-g_2(z)$.

(b) Pole/zero plot of $F(z)$. Plotted are the zeros of $1 - mg_2(z)$ ($\times$), the zeros of $1 - Lg_2(z)$ ($\circ$), the poles of $1 - H(z)$ ($\times$), and the zeros of $1 - H(z)$ ($\circ$).

(c) Phase plot of $(1 - Lg_2(z))/(1 - mg_2(z))$ (red), $1 - H(z)$ (green), and $F(z)$ (blue) for $z = \rho e^{j\theta}$. Note that $\angle F(z) \in [-\pi/2, \pi/2]$ for all $z \in \rho\mathbb{T}$ as desired.

Figure 6.7. Design of the parameters $(\alpha, \beta, \gamma, \rho)$ in the TM method.

the zero matrix. This small rank property is a known property of solutions to LMIs [**75**], and also holds for gradient descent with the optimal constant step size.

The design procedure is shown in Fig. 6.7. This does not guarantee global linear convergence, but gives some intuition for the design. Before analyzing the algorithm, we note that the design of both the TM method and gradient descent using the optimal step size have the following characteristics:

- The matrix in the LMI in (6.29) has rank zero.
- The real part of $F(z)$ in (6.44) is identically zero on $\rho\mathbb{T}$ (since the phase is $\pm\pi/2$).

6.5.2. Analysis (Proof of Theorem 22)

We now prove Theorem 22 using the IQC tools from Section 6.2. From Example 4, the TM method is equivalent to (6.18) with $\Delta = \nabla f$, $\eta_k = [\xi_k^T \; \xi_{k-1}^T]^T$, $p = 2n$, G given in (6.21), and the parameters in (6.7). Let $H(z) = \rho^2/z$ where $\rho = 1 - 1/\sqrt{\kappa}$. Then Lemma 21 gives $\nabla f \in \mathrm{IQC}(\Psi, M, u_\star, y_\star, \rho)$ where Ψ and M are given by (6.40) and $u_\star = \nabla f(y_\star)$.

Since $f(x)$ is strongly convex, it has a unique minimizer $x_\star$. The only point which is both a fixed point of the system and such that the gradient of f satisfies the IQC is

$$(u_\star, x_\star, y_\star, z_\star, \eta_\star, \zeta_\star) = \left(0, x_\star, x_\star, \begin{bmatrix} L(1-\rho^2)x_\star \\ -mx_\star \end{bmatrix}, \begin{bmatrix} x_\star \\ x_\star \end{bmatrix}, -Lx_\star\right). \tag{6.54}$$

In particular, the output of the algorithm is the minimizer of f when at this fixed point. Using the parameterization of Ψ in (6.41), the combined system $\hat{G}$ in (6.26) is

$$\left[\begin{array}{c|cc} \hat{A} & \hat{B}_1 & \hat{B}_2 \\ \hline \hat{C}_1 & \hat{D}_{11} & 0 \\ \hat{C}_2 & \hat{D}_{21} & \hat{D}_{22} \end{array}\right] = \left[\begin{array}{ccc|cc} 1+\beta & -\beta & 0 & 0 & -\alpha \\ 1 & 0 & 0 & 0 & 0 \\ -L(1+\gamma) & L\gamma & 0 & 0 & 1 \\ \hline 1+\delta & -\delta & 0 & 0 & 0 \\ L(1+\gamma) & -L\gamma & \rho^2 & 0 & -1 \\ -m(1+\gamma) & m\gamma & 0 & 0 & 1 \end{array}\right] \otimes I_n \tag{6.55}$$

where the state is $\hat{\xi} = [\eta^T,\ \zeta^T]^T$ with $\eta_k = [\xi_k^T,\ \xi_{k-1}^T]^T$. The system $\hat{G}$ is uncontrollable due to the pole/zero cancellation in $F(z)$ (see Fig. 6.7b). To make this explicit, let

$$\mathcal{C} = \begin{bmatrix} \hat{B} & \hat{A}\hat{B} & \hat{A}^2\hat{B} \end{bmatrix} \tag{6.56}$$

be the controllability matrix for $\hat{G}$. Since $\operatorname{rank}(\mathcal{C}) = 2 < 3$, the pair $(\hat{A}, \hat{B})$ is uncontrollable. We now apply a coordinate change to separate the uncontrollable state. Let

$$Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ -L(1+\epsilon) & L\epsilon & 1 \end{bmatrix} \tag{6.57}$$

where $\epsilon = -(1-\gamma/\beta)$. Note that the first two columns of Q form a basis for the column space of $\mathcal{C}$, and Q is nonsingular. Then applying the coordinate change $Q^{-1}\hat{\xi}$ gives a different parameterization of $\hat{G}$,

$$\hat{G} = \left[\begin{array}{ccc|cc} 1+\beta & -\beta & 0 & 0 & -\alpha \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \hline 1+\delta & -\delta & 0 & 0 & 0 \\ L(1+\gamma-\rho^2(1+\epsilon)) & -L(\gamma-\rho^2\epsilon) & \rho^2 & 0 & -1 \\ -m(1+\gamma) & m\gamma & 0 & 0 & 1 \end{array}\right] \otimes I_n, \tag{6.58}$$

from which it is clear that the third state is uncontrollable. Furthermore, for $k \geq 0$ the third state is zero and does not affect the output. Therefore, we define the reduced system

$$\bar{G} = \left[\begin{array}{cc|cc} 1+\beta & -\beta & 0 & -\alpha \\ 1 & 0 & 0 & 0 \\ \hline 1+\delta & -\delta & 0 & 0 \\ L(1+\gamma-\rho^2(1+\epsilon)) & -L(\gamma-\rho^2\epsilon) & 0 & -1 \\ -m(1+\gamma) & m\gamma & 0 & 1 \end{array}\right] \otimes I_n \tag{6.59}$$

which has state $\bar{\xi} = \eta$.

We now apply Theorem 23 to $\bar{G}$. Comparing (6.58) and (6.59), it is clear that (6.28) is satisfied with $k_0 = 1$ and $\bar{\xi}_0 = \eta_0 = [\xi_0^T \ \xi_{-1}^T]^T$. Letting $P = 2mL$, the right-hand side of (6.29) is the zero matrix, so the LMI in (6.29) is satisfied. Then for any initial condition $\xi_0, \xi_{-1} \in \mathbb{R}^n$, we have

$$\|x_k - x_\star\| \leq \rho^k \sqrt{\|x_0 - x_\star\|^2 + \frac{\nu}{2mL}} \tag{6.60}$$

for all $k \geq k_0 = 1$ where

$$\nu = \rho^{-2}\Big[(z_0 - z_\star)^T M(z_0 - z_\star) - (\bar{z}_0 - z_\star)^T M(\bar{z}_0 - z_\star)\Big].$$

From (6.55) and (6.59) and using $\zeta_0 = \zeta_\star$, we have

$$\begin{aligned} z_0 &= \begin{bmatrix} L(1+\gamma) & -L\gamma & \rho^2 \\ -m(1+\gamma) & m\gamma & 0 \end{bmatrix} \begin{bmatrix} \xi_0 \\ \xi_{-1} \\ \zeta_\star \end{bmatrix} \\ \bar{z}_0 &= \begin{bmatrix} L(1+\gamma-\rho^2(1+\epsilon)) & -L(\gamma-\rho^2\epsilon) \\ -m(1+\gamma) & m\gamma \end{bmatrix} \begin{bmatrix} \xi_0 \\ \xi_{-1} \end{bmatrix} \end{aligned}$$

which implies

$$\nu = 2L[\nabla f(y_0) - m(y_0 - x_\star)]^T[(1+\epsilon)\xi_0 - \epsilon\xi_{-1} - x_\star].$$

Using $\xi_{-1} = \xi_0$ and simplifying the bound in (6.60) gives the bound in (6.10). Applying Lemma 19 gives the bound in (6.11). ■

6.6. Simulations

Simulations are shown in Fig. 6.8 comparing (1) gradient descent using the optimal step size, (2) heavy-ball using the parameters in [**47**], (3) Nesterov's method with $\alpha = 1/L$ and $\beta = (\sqrt{L}-\sqrt{m})/(\sqrt{L}+\sqrt{m})$, and (4) the TM method. The objective functions are randomly generated piecewise quadratics with $n = 1$ similar to the heavy-ball counter-example in [**49**], where the coefficients are chosen such that $f \in \mathcal{S}_{1,50}$. The heavy-ball method contains stable limit cycles and does not converge globally. The TM method is both guaranteed to converge from any initial condition and faster than Nesterov's method. Simulation results for higher-dimensional piecewise quadratic objective functions are similar.

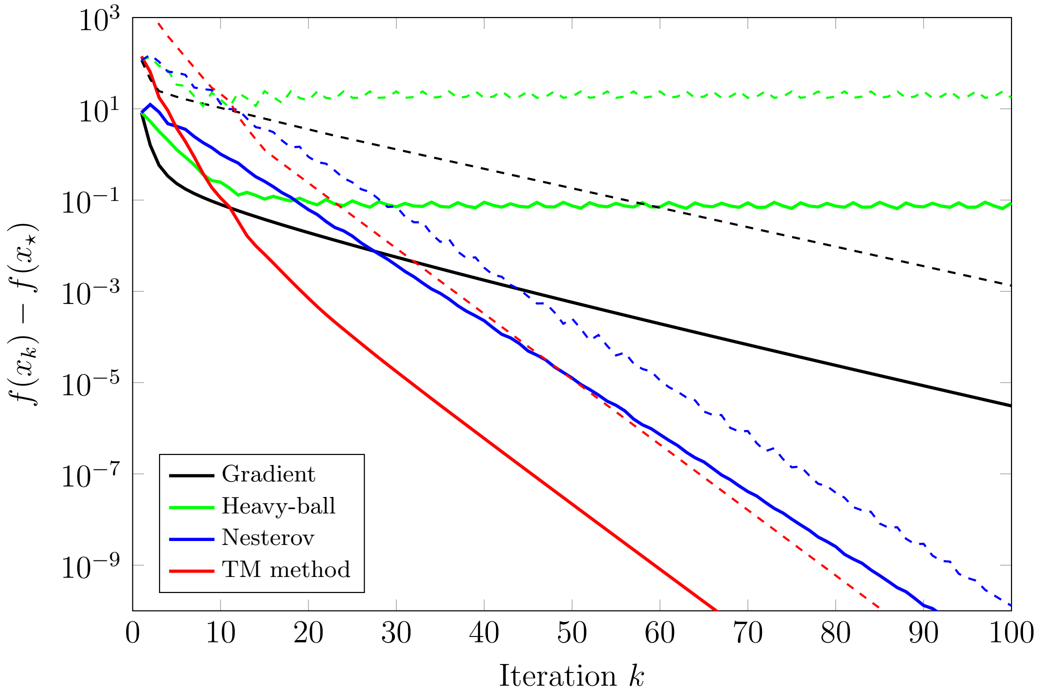


Figure 6.8. Simulation results of several optimization algorithms. The solid lines are the average and the dashed lines indicate the maximum over 1000 trials.

6.7. Alternative Convergence Proofs

We now provide alternative convergence proofs for both gradient descent and the triple momentum method which do not rely on control theory. First, we state our main analysis theorem which can be used to prove linear convergence of a sequence with rate ρ.

Theorem 24 (Analysis)**.** *Let $x \in \ell_{2e}^n$ and $q \in \ell_{2e}$ be arbitrary sequences, let $x_\star \in \mathbb{R}^n$, let $\mu \geq 0$, and let $k_0 \geq 0$. If*

$$\|x_{k+1} - x_\star\|^2 \leq \rho^2 \, \|x_k - x_\star\|^2 + \mu \, q_k, \quad \textit{for all } k \geq k_0 \tag{6.61}$$

and

$$0 \geq \sum_{j=k_0}^{k} \rho^{-2j} \, q_j, \quad \textit{for all } k \geq k_0, \tag{6.62}$$

then x_k converges to $x_\star$ linearly with rate ρ. Specifically,

$$\|x_k - x_\star\| \leq \rho^{k-k_0} \, \|x_{k_0} - x_\star\|, \quad \textit{for all } k \geq k_0. \tag{6.63}$$

Proof. Define the quanitity

$$\eta_k := \|x_{k+1} - x_\star\|^2 - \rho^2 \, \|x_k - x_\star\|^2 - \mu \, q_k. \tag{6.64}$$

From (6.61), we have $\eta_k \leq 0$ for all $k \geq k_0$. Then we have the following telescoping sum,

$$0 \geq \sum_{j=k_0}^{k-1} \rho^{-2(k-j-1)} \, \eta_j = \|x_k - x_\star\|^2 - \rho^{2(k-k_0)} \, \|x_{k_0} - x_\star\|^2 - \mu \, \rho^{2(k-1)} \sum_{j=k_0}^{k-1} \rho^{-2j} q_j \tag{6.65}$$

for all $k \geq k_0$. The summation term is non-positive from (6.62), so we have

$$\|x_k - x_\star\|^2 \leq \rho^{2(k-k_0)} \, \|x_{k_0} - x_\star\|^2 \tag{6.66}$$

for all $k \geq k_0$. Taking the square root then gives (6.63). □

Remark 15. *The following are some examples of when* (6.62) *holds:*

- *$q_k \leq 0$ for all $k \geq 0$.*

- *There exists a sequence $\bar{q} \in \ell_{2e}$ with $\bar{q}_k \le 0$ for all $k \ge 0$ such that*

$$q_k = \begin{cases} \bar{q}_k - \rho^2\, \bar{q}_{k-1}, & k \ge 1 \\ \bar{q}_0, & k = 0. \end{cases} \tag{6.67}$$

In order to use Theorem 24, we need a sequence q_k that satisfies (6.62) and enables us to show that (6.61) holds. When the sequence x_k is generated by gradient algorithms applied to a strongly convex function, we use the following lemmas to characterize the gradient of functions in $\mathcal{S}_{m,L}$.

Lemma 23. *Suppose $f \in \mathcal{S}_{m,L}$. Then*

$$0 \ge \big[\nabla f(x) - \nabla f(y) - m\,(x-y)\big]^T \big[\nabla f(x) - \nabla f(y) - L\,(x-y)\big] \tag{6.68}$$

for all $x, y \in \mathbb{R}^n$.

Proof. The result follows directly from the quadratic inequality in [**49**, Lemma 6]. □

Lemma 24. *Suppose $f \in \mathcal{S}_{m,L}$ with $0 = \nabla f(x_\star)$. Define $p_r(y) := \nabla f(y) - r\,(y - x_\star)$. Given a sequence $y \in \ell_{2e}^n$, let*

$$q_k = \begin{cases} p_m(y_k)^T \big[p_L(y_k) - \rho^2\, p_L(y_{k-1})\big], & k \ge 1 \\ p_m(y_0)^T p_L(y_0), & k = 0. \end{cases} \tag{6.69}$$

Then

$$0 \ge \sum_{j=0}^{k} \rho^{-2j}\, q_j, \quad \textit{for all } k \ge 0. \tag{6.70}$$

Proof. The result follows directly from the quadratic inequality in [**49**, Lemma 10]. □

We now apply Theorem 24 along with Lemmas 23 and 24 to prove that the sequences x_k produced by gradient descent and the triple momentum method converge linearly to the optimizer $x_\star$ with rate ρ. We prove the bound for gradient descent using the optimal step size in Theorem 20 which we restate here.

Theorem 25 (Gradient descent)**.** *Let $f_k \in \mathcal{S}_{m,L}$ for all $k \geq 0$ with $0 < m \leq L$. Let $x_\star \in \mathbb{R}^n$ be the unique minimizer of f_k, i.e., $0 = \nabla f_k(x_\star)$ for all $k \geq 0$. For any initial condition $\xi_0 \in \mathbb{R}^n$, gradient descent with step size $\alpha = 2/(L+m)$ produces iterates which satisfy*

$$\|x_k - x_\star\| \leq \rho^k \|x_0 - x_\star\| \tag{6.71}$$

$$f(x_k) - f(x_\star) \leq \rho^{2k} \frac{L}{2} \|x_0 - x_\star\|^2 \tag{6.72}$$

for all $k \geq 0$ where $\rho = (L-m)/(L+m)$.

Proof. Let x_k be the sequence generated by gradient descent with initial condition ξ_0. Define the sequence

$$q_k = \big[\nabla f_k(x_k) - m\,(x_k - x_\star)\big]^T \big[\nabla f_k(x_k) - L\,(x_k - x_\star)\big]. \tag{6.73}$$

Then $q_k \leq 0$ for all $k \geq 0$ from Lemma 23, so (6.62) holds. Next, we show that (6.61) holds for all $k \geq 0$. Define $\nabla f_k := \nabla f_k(x_k)$. Then for $k \geq 0$, we have

$$\begin{aligned}\eta_k &:= \|x_{k+1} - x_\star\|^2 - \rho^2 \|x_k - x_\star\|^2 - \mu\, q_k \\ &= \|x_{k+1} - x_\star\|^2 - \rho^2 \|x_k - x_\star\|^2 - \mu \big[\nabla f_k - m\,(x_k - x_\star)\big]^T \big[\nabla f_k - L\,(x_k - x_\star)\big].\end{aligned}$$

Making the substitution $x_{k+1} \to x_k - \alpha \nabla f_k$ gives η_k in terms of x_k, ∇f_k, and $x_\star$. Using the parameters

$$(\rho, \mu, \alpha) = \left(\frac{L-m}{L+m},\ \frac{4}{(L+m)^2},\ \frac{2}{L+m} \right)$$

gives $\eta_k \equiv 0$ for any $x_k, \nabla f_k, x_\star \in \mathbb{R}^n$. (In fact, these parameters are the unique solution with $\rho \in [0,1)$ to the equation $\eta_k \equiv 0$.) Then (6.61) is satisfied with equality for $k \geq 0$. Applying Theorem 24 gives $\|x_k - x_\star\| \leq \rho^k \|x_0 - x_\star\|$ for all $k \geq 0$, and Lemma 19 gives the bound on the function values. □

Theorem 26 (Triple momentum method). *Let $f \in \mathcal{S}_{m,L}$ with $0 < m \leq L$. Let $x_\star \in \mathbb{R}^n$ be the unique minimizer of f. For any initial condition $\xi_0, \xi_{-1} \in \mathbb{R}^n$, the TM method produces iterates which satisfy*

$$\|x_k - x_\star\| \leq \rho^{k-1} \|x_1 - x_\star\| \tag{6.74}$$

$$f(x_k) - f(x_\star) \leq \rho^{2(k-1)} \frac{L}{2} \|x_1 - x_\star\|^2 \tag{6.75}$$

for all $k \geq 1$.

Proof. Let x_k be the sequence generated by the TM method with initial conditions ξ_0 and ξ_{-1}. Define the sequence

$$q_k = \begin{cases} p_m(y_k)^T \big[p_L(y_k) - \rho^2\, p_L(y_{k-1})\big], & k \geq 1 \\ p_m(y_1)^T p_L(y_1), & k = 0. \end{cases} \tag{6.76}$$

Then (6.62) holds for all $k \geq 1$ from Lemma 24. Next, we show that (6.61) holds for $k \geq 1$. Define $\nabla f_k := \nabla f(y_k)$. Then for $k \geq 1$, we have

$$\begin{aligned}\eta_k &:= \|x_{k+1} - x_\star\|^2 - \rho^2 \|x_k - x_\star\|^2 - \mu\, q_k \\ &= \|x_{k+1} - x_\star\|^2 - \rho^2 \|x_k - x_\star\|^2 \\ &\quad - \mu \left[\nabla f_k - m\,(y_k - x_\star)\right]^T \Big[\big(\nabla f_k - L\,(y_k - x_\star)\big) - \rho^2 \big(\nabla f_{k-1} - L\,(y_{k-1} - x_\star)\big)\Big].\end{aligned}$$

Making the substitutions

$$\begin{aligned} x_{k+1} &\to (1+\delta)\xi_{k+1} - \delta\xi_k \\ x_k &\to (1+\delta)\xi_k - \delta\xi_{k-1} \\ y_k &\to (1+\gamma)\xi_k - \gamma\xi_{k-1} \\ y_{k-1} &\to (1+\gamma)\xi_{k-1} - \gamma\xi_{k-2} \\ \nabla f_k &\to \big[-\xi_{k+1} + (1+\beta)\xi_k - \beta\xi_{k-1}\big]/\alpha \\ \nabla f_{k-1} &\to \big[-\xi_k + (1+\beta)\xi_{k-1} - \beta\xi_{k-2}\big]/\alpha \end{aligned}$$

gives η_k in terms of ξ_{k-2}, ξ_{k-1}, ξ_k, ξ_{k+1}, and $x_\star$. Using the parameters

$$(\rho, \mu, \alpha, \beta, \gamma, \delta) = \left(1 - \sqrt{\frac{m}{L}},\ \frac{1}{mL},\ \frac{1+\rho}{L},\ \frac{\rho^2}{2-\rho},\ \frac{\rho^2}{(1+\rho)(2-\rho)},\ \frac{\rho^2}{1-\rho^2}\right)$$

gives $\eta_k \equiv 0$ for any $\xi_{k-2}, \xi_{k-1}, \xi_k, \xi_{k+1}, x_\star \in \mathbb{R}^n$ for $k \geq 1$. (In fact, these parameters are the unique solution with $\rho \in [0, 1)$ to the equation $\eta_k \equiv 0$.) Then (6.61) is satisfied with equality for $k \geq 1$. Applying Theorem 24 gives $\|x_k - x_\star\| \leq \rho^{k-1} \|x_1 - x_\star\|$ for all $k \geq 1$, and Lemma 19 gives the bound on the function values. □

6.8. Summary

We have both designed and analyzed the TM method for gradient-based optimization. When $f \in \mathcal{S}_{m,L}$, the iterates converge linearly to the optimizer at rate $\rho = 1 - \sqrt{m/L}$ from any initial condition, and the function values converge at rate ρ^2. This is the fastest known convergence rate that has been proven for first-order algorithms which converge globally to the minimizer. For high levels of accuracy, the bound on the iteration complexity for our algorithm is half the known bound for Nesterov's method and within a factor of two of the theoretical lower bound in [**48**, Thm. 2.1.12].

Further analysis is needed to study the behavior of the TM method in the case when f is weakly convex and to characterize the robustness to the parameters m and L.

The IQC framework used to design and analyze our algorithm can be expanded to handle many different problems and methods such as proximal algorithms, inexact gradient methods, and optimizing weakly convex functions; see [**49**, Sections 5 and 7] for details. Future work will focus on using our design techniques to design optimal methods in such scenarios.

CHAPTER 7

Conclusion

7.1. Summary

In this thesis, we have analyzed and designed algorithms for two problems: (1) the dynamic average consensus problem, and (2) the problem of minimizing a strongly convex function using only gradient information.

The dynamic average consensus problem was separated into two cases depending on whether the frequency spectrum of the input signals is composed of discrete frequencies or a continuous band of frequencies, and estimators were designed for each case. Various feedback estimators were proposed, each with its own benefits and drawbacks, and each estimator achieves zero steady-state error when the model of the input is known. Zero steady-state error is not possible when the input signals are bandlimited, but a feedforward estimator was designed which achieves arbitrarily small steady-state error even when the communication graph changes at each iteration.

We have also designed a novel gradient-based algorithm for minimizing a convex function. When the function is smooth and strongly convex, our algorithm is the fastest known globally convergent first-order method for minimizing the function. Our algorithm generalizes well-known methods such as gradient descent, Nesterov's accelerated gradient descent, and the heavy-ball method.

Although the two problems studied have vastly different applications, we have found a connection between dynamic average consensus and convex optimization. Estimators for dynamic average consensus can be viewed as applying gradient-based convex optimization algorithms to a specific function whose gradient is the graph Laplacian. This allows the *centralized* optimization algorithm to be implemented in a *distributed* manner.

7.2. Future Directions

There are many exciting areas where our work could be expanded. We list a few below.

- When the input signals and graph are constant, it is an open question whether there exists an estimator which has all of the following properties: 1) scalable, 2) exact, 3) robust to initial conditions, 4) time-invariant, 5) internally stable, 6) convergence rate $\rho = (1-\sqrt{\lambda_r})/(1+\sqrt{\lambda_r})$, and 7) one-hop local broadcast communication. Each of the feedback estimators in Chapter 4 lacks at least one of these properties. We now give several approaches to solving this problem.
  - ○ The PI estimator has all of the properties except for the optimal convergence rate (although the convergence rate of the PI-4 estimator is comparable). Therefore, one approach is to introduce more degrees of freedom into the PI estimator. Both the proportional and integral terms in the PI estimator get filtered through $h_1(z)$. This was needed to design the estimator using the nested root locus technique. However, a more general approach would be to use different filters as shown in Fig. 7.1. The cost of using different filters is that this estimator requires each agent to transmit three variables per iteration, but the benefit is that the convergence rate is potentially faster. It is an open question how to design such an estimator to optimize the convergence rate, and what the resulting optimal convergence rate is.

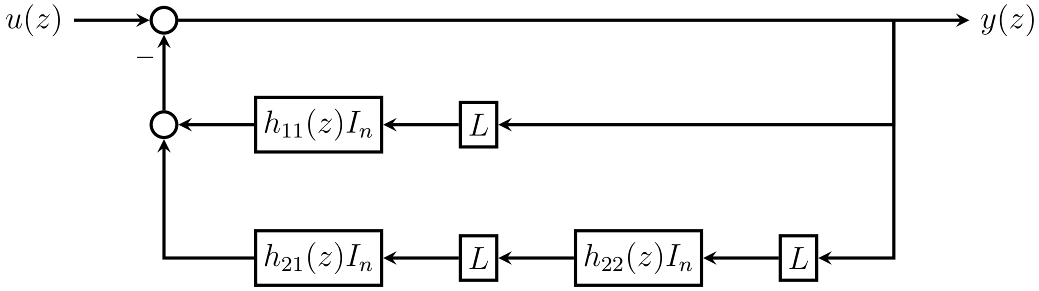
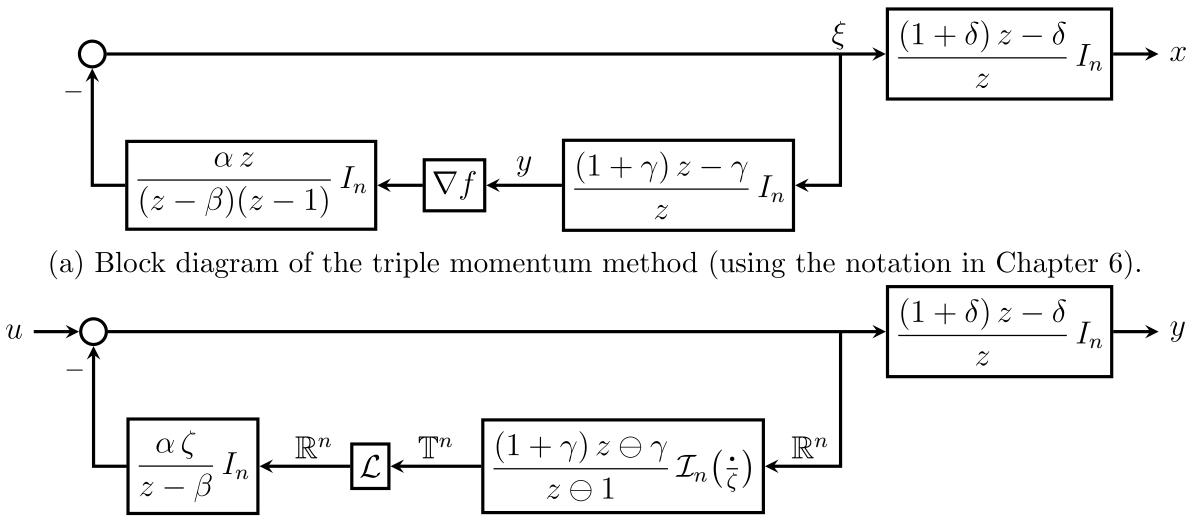


Figure 7.1. Block diagram of a generalized proportional-integral estimator.

- ◦ Another approach is to modify the nonlinear estimator to achieve global convergence. Estimators can be viewed as applying gradient-based optimization algorithms to the scalar function for which the Laplacian is the gradient. The NL-2 estimator is comparable to using the heavy-ball method for the optimization. It is known that the heavy-ball method is not globally convergent for strongly convex functions, and neither is the resulting estimator. From Chapter 6, however, we have a gradient-based optimization method which *is* globally convergent for strongly convex functions. We would like to show that the corresponding estimator is also globally convergent.

  The results from Chapter 6 cannot be directly applied to the nonlinear estimator, however. To see this, consider the block diagram of the triple momentum method and the corresponding estimator in Fig. 7.2. The nonlinear estimator is similar to the triple momentum method applied to the function $\mathcal{L} \circ \mathcal{I}$, except that (1) the estimator has an input u, and (2) the integrator is moved *between* the Laplacian operator $\mathcal{L}$ and the projection operator $\mathcal{I}$ so that it operates on the compact set $\mathbb{T}^n$. It is an open question whether the resulting estimator is globally convergent, and how large the scaling parameter ζ must be chosen to guarantee convergence.


ξ
$\frac{(1+\delta)\,z-\delta}{z}\,I_n$
x
$-$
$\frac{\alpha\,z}{(z-\beta)(z-1)}\,I_n$
∇f
y
$\frac{(1+\gamma)\,z-\gamma}{z}\,I_n$
u
$\frac{(1+\delta)\,z-\delta}{z}\,I_n$
y
$-$
$\frac{\alpha\,\zeta}{z-\beta}\,I_n$
$\mathbb{R}^n$
$\mathcal{L}$
$\mathbb{T}^n$
$\frac{(1+\gamma)\,z\ominus\gamma}{z\ominus 1}\,\mathcal{I}_n\big(\frac{\cdot}{\zeta}\big)$
$\mathbb{R}^n$

(a) Block diagram of the triple momentum method (using the notation in Chapter 6).

(b) Block diagram of the nonlinear estimator based on the triple momentum method.

Figure 7.2. Block diagram of the proposed nonlinear estimator based on the triple momentum method.

- The feedforward estimator in Chapter 5 was designed for bandlimited input signals with known cutoff frequency $\theta_c < \pi$. An interesting question is how to design the prefilter to minimize the error when the magnitude of the frequency spectrum is an arbitrary function $f(\theta)$, i.e., $f(\theta) = |u(e^{j\theta})|$ is known for all $\theta \in [0,\pi]$. We solved for the prefilter in the case

  $$f(\theta) = \begin{cases} 1, & \theta \in [0,\theta_c] \\ 0, & \text{otherwise.} \end{cases}$$

  This is a filter design problem where the prefilter $h_{\text{pre}}(z)$ must be designed to approximate z^ℓ in both magnitude *and* phase with weighting function $f(\theta)$.

- We characterized the convergence properties of the triple momentum method in Chapter 6 when the objective function is strongly convex and its gradient is Lipschitz continuous. In that case, we showed that the TM method converges approximately twice as fast as Nesterov's method. An algorithm has also been developed for the case when the objective function is weakly convex, and it has been shown to converge faster than Nesterov's method [**53**]. We need to study the TM method in the case when the objective function is weakly convex. If the convergence rate is slower than that in [**53**], an interesting question is whether there exists a single algorithm which achieves the best known convergence rate in both cases of strongly and weakly convex objective functions.

- The triple momentum method is also limited to solving unconstrained optimization problems. A common method of generalizing gradient methods to solve constrained problems is using the proximal operator (e.g., the proximal gradient method). The integral quadratic constraint tools can also be applied to such algorithms [**49**]. Therefore, we would like to design and analyze a proximal version of the triple momentum method for problems with constraints.

References

[1] R. Olfati-Saber, “Distributed Kalman filter with embedded consensus filters,” in *Proc. of the Joint 44th IEEE Conf. on Decision and Control and 2005 European Control Conf.*, 2005, pp. 8179–8184.

[2] ——, “Distributed Kalman filtering for sensor networks,” in *Proc. of the 46th IEEE Conf. on Decision and Control*, 2007, pp. 5492–5498.

[3] J. Cortés, “Distributed Kriged Kalman filter for spatial estimation,” *IEEE Trans. Autom. Control*, vol. 54, no. 12, pp. 2816–2827, 2009.

[4] K. Lynch, I. Schwartz, P. Yang, and R. Freeman, “Decentralized environmental modeling by mobile sensor networks.” in *IEEE Trans. Robot.*, 2008.

[5] H. Bai, R. Freeman, and K. Lynch, “Distributed Kalman filtering using the internal model average consensus estimator,” in *Proc. of the 2011 Amer. Control Conf.*, 2011, pp. 1500–1505.

[6] C. K. Peterson and D. A. Paley, “Distributed estimation for motion coordination in an unknown spatially varying flowfield,” *Journal of Guidance, Control, and Dynamics*, vol. 36, no. 3, pp. 894–898, 2013.

[7] R. Olfati-Saber and J. S. Shamma, “Consensus filters for sensor networks and distributed sensor fusion,” in *Proc. of the 44th IEEE Conf. on Decision and Control*, Dec. 2005, pp. 6698–6703.

[8] P. Yang, R. Freeman, and K. Lynch, “Multi-agent coordination by decentralized estimation and control,” *IEEE Trans. Autom. Control*, vol. 53, no. 11, pp. 2480–2496, 2008.

[9] R. Aragüés, J. Cortés, and C. Sagüés, “Distributed consensus on robot networks for dynamically merging feature-based maps,” *IEEE Trans. Robot.*, vol. 28, no. 4, pp. 840–854, 2012.

[10] G. Qu and N. Li, “Harnessing smoothness to accelerate distributed optimization,” in *Proc. of the 55th IEEE Conf. on Decision and Control*, 2016, pp. 159–166.

[11] J. Tsitsiklis, "Problems in decentralized decision making and computation," Ph.D. dissertation, Massachusetts Institute of Technology, Nov. 1984.

[12] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," in *Proc. of the 42nd IEEE Conf. on Decision and Control*, vol. 5, Dec. 2003, pp. 4997–5002.

[13] B. N. Oreshkin, M. J. Coates, and M. G. Rabbat, "Optimization and analysis of distributed averaging with short node memory," *IEEE Trans. Signal Process.*, vol. 58, no. 5, pp. 2850–2865, May 2010.

[14] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast consensus by the alternating direction multipliers method," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5523–5537, Nov. 2011.

[15] R. Freeman, T. Nelson, and K. Lynch, "A complete characterization of a class of robust linear average consensus protocols," in *Proc. of the 2010 Amer. Control Conf.*, 2010, pp. 3198–3203.

[16] D. P. Spanos, R. Olfati-Saber, and R. M. Murray, "Dynamic consensus on mobile networks," *IFAC World Congress*, pp. 1–6, 2005.

[17] S. Nosrati, M. Shafiee, and M. B. Menhaj, "Dynamic average consensus via nonlinear protocols," *Automatica*, vol. 48, no. 9, pp. 2262 – 2270, 2012.

[18] F. Chen, Y. Cao, and W. Ren, "Distributed average tracking of multiple time-varying reference signals with bounded derivatives," *IEEE Trans. Autom. Control*, vol. 57, no. 12, pp. 3169–3174, Dec. 2012.

[19] R. Freeman, P. Yang, and K. Lynch, "Stability and convergence properties of dynamic average consensus estimators," in *Proc. of the 45th IEEE Conf. on Decision and Control*, 2006, pp. 338–343.

[20] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "Design of robust dynamic average consensus estimators," in *Proc. of the 54th IEEE Conf. on Decision and Control*, Dec. 2015, pp. 6269–6275.

[21] H. Bai, R. Freeman, and K. Lynch, "Robust dynamic average consensus of time-varying inputs," in *Proc. of the 49th IEEE Conf. on Decision and Control*, 2010, pp. 3104–3109.

[22] H. Bai, "Adaptive motion coordination with an unknown reference velocity," in *Proc. of the 2015 Amer. Control Conf.*, July 2015, pp. 5581–5586.

[23] M. Zhu and S. Martínez, "Discrete-time dynamic average consensus," *Automatica*, vol. 46, no. 2, pp. 322 – 329, 2010.

[24] S. Kia, J. Cortés, and S. Martínez, "Dynamic average consensus under limited control authority and privacy requirements," *International Journal of Robust and Nonlinear Control*, 2013.

[25] M. Franceschelli and A. Gasparri, "Multi-stage discrete time dynamic average consensus," in *Proc. of the 55nd IEEE Conf. on Decision and Control*, Dec. 2016.

[26] Y. Yuan, J. Liu, R. M. Murray, and J. Gonçalves, "Decentralised minimal-time dynamic consensus," in *Proc. of the 2012 Amer. Control Conf.*, June 2012, pp. 800–805.

[27] E. Montijano, J. I. Montijano, C. Sagüés, and S. Martínez, "Robust discrete time dynamic average consensus," *Automatica*, vol. 50, no. 12, pp. 3131 – 3138, 2014.

[28] E. Kokiopoulou and P. Frossard, "Polynomial filtering for fast convergence in distributed consensus," *IEEE Trans. Signal Process.*, vol. 57, no. 1, pp. 342–354, Jan. 2009.

[29] E. Montijano, J. Montijano, and C. Sagüés, "Chebyshev polynomials in distributed consensus applications," *IEEE Trans. Signal Process.*, vol. 61, no. 3, pp. 693–706, Feb. 2013.

[30] M. L. Elwin, R. A. Freeman, and K. M. Lynch, "A systematic design process for internal model average consensus estimators," in *Proc. of the 52nd IEEE Conf. on Decision and Control*, Dec. 2013, pp. 5878–5883.

[31] ——, "Worst-case optimal average consensus estimators for robot swarms," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sept. 2014, pp. 3814–3819.

[32] K. Cai and H. Ishii, "Average consensus on arbitrary strongly connected digraphs with dynamic topologies," in *Proc. of the 2012 Amer. Control Conf.*, 2012, pp. 14–19.

[33] N. Vaidya, C. Hadjicostis, and A. Dominguez-Garcia, "Robust average consensus over packet dropping links: Analysis via coefficients of ergodicity," in *Proc. of the 51st IEEE Conf. on Decision and Control*, 2012, pp. 2761–2766.

[34] J.-Y. Chen and J. Hu, "On the convergence of distributed random grouping for average consensus on sensor networks with time-varying graphs," in *Proc. of the 46th IEEE Conf. on Decision and Control*, 2007, pp. 4233–4238.

[35] Y. Chen, R. Tron, A. Terzis, and R. Vidal, "Corrective consensus: Converging to the exact average," in *Proc. of the 49th IEEE Conf. on Decision and Control*, 2010, pp. 1221–1228.

[36] T. Li and J.-F. Zhang, "Consensus conditions of multi-agent systems with time-varying topologies and stochastic communication noises," *IEEE Trans. Autom. Control*, vol. 55, no. 9, pp. 2043–2057, 2010.

[37] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. on Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sept. 2004.

[38] U. A. Khan, "High dimensional consensus in large-scale networks: Theory and applications," Ph.D. dissertation, Carnegie Mellon University, 2009.

[39] P. P. Vaidyanathan, "On predicting a band-limited signal based on past sample values," *Proc. of the IEEE*, vol. 75, no. 8, pp. 1125–1127, Aug. 1987.

[40] Y. Mo and R. M. Murray, "Privacy preserving average consensus," *IEEE Trans. on Autom. Control*, vol. 62, no. 2, pp. 753–765, Feb. 2017.

[41] A. D. Dominguez-Garcia and C. N. Hadjicostis, "Distributed matrix scaling and application to average consensus in directed graphs," *IEEE Trans. on Autom. Control*, vol. 58, no. 3, pp. 667–681, 2013.

[42] N. E. Manitara and C. N. Hadjicostis, "Distributed stopping for average consensus in digraphs," *IEEE Trans. on Control of Network Systems*, vol. PP, no. 99, pp. 1–1, 2017.

[43] A. Kashyap, T. Başar, and R. Srikant, "Quantized consensus," *Automatica*, vol. 43, no. 7, pp. 1192 – 1203, 2007.

[44] T. C. Aysal, M. J. Coates, and M. G. Rabbat, "Distributed average consensus with dithered quantization," *IEEE Trans. Signal Process.*, vol. 56, no. 10, pp. 4905–4918, Oct. 2008.

[45] C. Nowzari and J. Cortés, "Zeno-free, distributed event-triggered communication and control for multi-agent average consensus," in *Proc. of the 2014 Amer. Control Conf.*, June 2014, pp. 2148–2153.

[46] Z. Sun, N. Huang, B. D. O. Anderson, and Z. Duan, "A new distributed zeno-free event-triggered algorithm for multi-agent consensus," in *Proc. of the 55th IEEE Conf. on Decision and Control*, Dec. 2016, pp. 3444–3449.

[47] B. Polyak, "Some methods of speeding up the convergence of iteration methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 4, no. 5, pp. 1 – 17, 1964.

[48] Y. Nesterov, *Introductory lectures on convex optimization: A basic course.* Springer, 2004.

[49] L. Lessard, B. Recht, and A. Packard, "Analysis and design of optimization algorithms via integral quadratic constraints," *SIAM Journal on Optimization*, vol. 26, no. 1, pp. 57–95, 2016.

[50] E. Ghadimi, H. R. Feyzmahdavian, and M. Johansson, "Global convergence of the heavy-ball method for convex optimization," in *2015 European Control Conference*, July 2015, pp. 310–315.

[51] A. Botev, G. Lever, and D. Barber, "Nesterov's accelerated gradient and momentum as approximations to regularised update descent," *ArXiv e-prints*, July 2016.

[52] Y. Drori and M. Teboulle, "Performance of first-order methods for smooth convex minimization: A novel approach," *Math. Program.*, vol. 145, no. 1, pp. 451–482, 2014.

[53] D. Kim and J. A. Fessler, "Optimized first-order methods for smooth convex minimization," *Math. Program.*, vol. 159, no. 1-2, pp. 81–107, Sept. 2016.

[54] E. Mallada, R. A. Freeman, and A. K. Tang, "Distributed synchronization of heterogeneous oscillators on networks with arbitrary topology," *IEEE Trans. on Control of Network Systems*, vol. 3, no. 1, pp. 12–23, 2016.

[55] P. Yang, R. Freeman, and K. Lynch, "Optimal information propagation in sensor networks," in *Proc. of the 2006 IEEE Int. Conf. on Robotics and Automation*, 2006, pp. 3122–3127.

[56] S. You, "A fast linear consensus protocol on an asymmetric directed graph," in *Proc. of the 2014 Amer. Control Conf.*, June 2014, pp. 3281–3286.

[57] J. M. Hendrickx and J. N. Tsitsiklis, "Fundamental limitations for anonymous distributed systems with broadcast communications," in *Proc. of the 53rd Allerton Conf. on Commun., Control, and Computing*, Sept. 2015, pp. 9–16.

[58] P. J. Basser and S. Pajevic, "Spectral decomposition of a 4th-order covariance tensor: Applications to diffusion tensor MRI," *Signal Processing*, vol. 87, no. 2, pp. 220 – 236, 2007.

[59] H. Neudecker, "Some theorems on matrix differentiation with special reference to Kronecker matrix products," *Journal of the American Statistical Association*, vol. 64, no. 327, pp. 953–963, 1969.

[60] A. Leon-Garcia, *Probability, Statistics, and Random Processes for Electrical Engineering.* Pearson/Prentice Hall, 2008.

[61] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "Asymptotic mean ergodicity of average consensus estimators," in *Proc. of the 2014 Amer. Control Conf.*, June 2014, pp. 4696–4701.

[62] E. Montijano, J. I. Montijano, C. Sagues, and S. Martínez, "Step size analysis in discrete-time dynamic average consensus," in *Proc. of the 2014 Amer. Control Conf.*, June 2014, pp. 5127–5132.

[63] B. Wellman and J. Hoagg, "Quadratically parameterized root locus analysis," *IEEE Trans. Autom. Control*, vol. 59, no. 7, pp. 1803–1817, July 2014.

[64] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "Optimal worst-case dynamic average consensus," in *Proc. of the 2015 Amer. Control Conf.*, July 2015, pp. 5324–5329.

[65] D. Henrion, D. Peaucelle, D. Arzelier, and M. Sebek, "Ellipsoidal approximation of the stability domain of a polynomial," *IEEE Trans. Autom. Control*, vol. 48, no. 12, pp. 2255–2259, Dec. 2003.

[66] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "Feedforward estimators for the distributed average tracking of bandlimited signals in discrete time with switching graph topology," in *Proc. of the 55th IEEE Conf. on Decision and Control*, Dec. 2016, pp. 4284–4289.

[67] W. Splettstösser, "On the prediction of band-limited signals from past samples," *Information Sciences*, vol. 28, no. 2, pp. 115 – 130, 1982.

[68] C. Bardaro, P. L. Butzer, R. L. Stens, and G. Vinti, "Prediction by samples from the past with error estimates covering discontinuous signals," *IEEE Trans. on Inf. Theory*, vol. 56, no. 1, pp. 614–633, Jan. 2010.

[69] D. H. Mugler, Y. Wu, and S. Clary, "Linear prediction of bandpass signals based on past samples," in *Proc. of the International Workshop on Sampling Theory*, Aug. 1999, pp. 119–124.

[70] N. Dokuchaev, "Predictors for discrete time processes with energy decay on higher frequencies," *IEEE Trans. Signal Process.*, vol. 60, no. 11, pp. 6027–6030, Nov. 2012.

[71] A. Rantzer, "On the Kalman-Yakubovich-Popov lemma," *Syst. Control Lett.*, vol. 28, no. 1, pp. 7–10, June 1996.

[72] P. Seiler, "Stability analysis with dissipation inequalities and integral quadratic constraints," *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1704–1709, June 2015.

[73] K. H. Khalil, *Nonlinear Systems*, 2nd ed. Prentice-Hall, Inc., 1996.

[74] N. S. Ahmad, J. Carrasco, and W. P. Heath, “A less conservative LMI condition for stability of discrete-time systems with slope-restricted nonlinearities,” *IEEE Trans. Autom. Control*, vol. 60, no. 6, pp. 1692–1697, June 2015.

[75] D. Henrion, S. Naldi, and M. S. E. Din, “Exact algorithms for linear matrix inequalities,” *SIAM Journal on Optimization*, vol. 26, no. 4, pp. 2512–2539, 2016.

Vita

Bryan R. Van Scoy was born in 1988 in Ashland, Ohio. He received B.S. and M.S. degrees in applied mathematics along with a B.S.E. in electrical engineering from the University of Akron in 2012.

Publications

[P1] B. Van Scoy, R. A. Freeman, and K. M. Lynch, "Asymptotic mean ergodicity of average consensus estimators," in *Proc. of the 2014 Amer. Control Conf.*, June 2014, pp. 4696–4701.

[P2] ——, "Optimal worst-case dynamic average consensus," in *Proc. of the 2015 Amer. Control Conf.*, July 2015, pp. 5324–5329.

[P3] ——, "Exploiting memory in dynamic average consensus," in *Proc. of the 53rd Annual Allerton Conf. on Comm., Control, and Computing*, Sept. 2015, pp. 258–265.

[P4] ——, "A fast robust nonlinear dynamic average consensus estimator in discrete time," in *Proc. of the 5th IFAC Workshop on Distributed Estimation and Control in Networked Systems*, vol. 48, no. 22, 2015, pp. 191 – 196.

[P5] ——, "Design of robust dynamic average consensus estimators," in *Proc. of the 54th IEEE Conf. on Decision and Control*, Dec. 2015, pp. 6269–6275.

[P6] ——, "Feedforward estimators for the distributed average tracking of bandlimited signals in discrete time with switching graph topology," in *Proc. of the 55th IEEE Conf. on Decision and Control*, Dec. 2016, pp. 4284–4289.