Distributed Averaging and Balancing in Network Systems

with Applications to Coordination and Control
Other titles in Foundations and Trends® in Systems and Control

Nonlinear Model Reduction by Moment Matching
Giordano Scarciotti and Alessandro Astolfi
ISBN: 978-1-68083-330-0

Logical Control of Complex Resource Allocation Systems
Spyros Reveliotis
ISBN: 978-1-68083-250-1

Observability of Hybrid Dynamical Systems
Elena De Santis and Maria Domenica Di Benedetto
ISBN: 978-1-68083-220-4

Operator Splitting Methods in Control
Giorgos Stathopoulos, Harsh Shukla, Alexander Szucs, Ye Pu and Colin N. Jones
ISBN: 978-1-68083-174-0

Sensor Fault Diagnosis
Vasso Reppa, Marios M. Polycarpou and Christos G. Panayiotou
Distributed Averaging and Balancing in Network Systems
with Applications to Coordination and Control

Christoforos N. Hadjicostis
University of Cyprus
chadjic@ucy.ac.cy

Alejandro D. Domínguez-García
University of Illinois
aledan@illinois.edu

Themistoklis Charalambous
Aalto University
themistoklis.charalambous@aalto.fi
# Editors-in-Chief

<table>
<thead>
<tr>
<th>Panos J. Antsaklis</th>
<th>Alessandro Astolfi</th>
</tr>
</thead>
<tbody>
<tr>
<td>University of Notre Dame</td>
<td>Imperial College London, United Kingdom</td>
</tr>
<tr>
<td>United States</td>
<td>University of Rome &quot;Tor Vergata&quot;, Italy</td>
</tr>
</tbody>
</table>

# Editors

<table>
<thead>
<tr>
<th>John Baillieuli</th>
<th>Dragan Nesic</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Boston University</em></td>
<td><em>The University of Melbourne</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Peter Caines</th>
<th>Marios Polycarpou</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>McGill University</em></td>
<td><em>University of Cyprus</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Christos Cassandras</th>
<th>Jörg Raisch</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Boston University</em></td>
<td><em>Technical University Berlin</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Denis Dochain</th>
<th>Arjan van der Schaft</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>UC Louvain</em></td>
<td><em>University of Groningen</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Magnus Egerstedt</th>
<th>M. Elena Valcher</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>Georgia Institute of Technology</em></td>
<td><em>University of Padova</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Karl Henrik Johansson</th>
<th>Richard Vinter</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>KTH Stockholm</em></td>
<td><em>Imperial College London</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Miroslav Krstic</th>
<th>George Weiss</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>University of California, San Diego</em></td>
<td><em>Tel Aviv University</em></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Jan Maciejewski</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>University of Cambridge</em></td>
</tr>
</tbody>
</table>
Editorial Scope

Topics

Foundations and Trends® in Systems and Control publishes survey and tutorial articles in the following topics:

• Control of:
  – Hybrid and Discrete Event Systems
  – Nonlinear Systems
  – Network Systems
  – Stochastic Systems
  – Multi-agent Systems
  – Distributed Parameter Systems

• Delay Systems
  – Filtering, Estimation, Identification
  – Optimal Control
  – Systems Theory
  – Control Applications

Information for Librarians

Foundations and Trends® in Systems and Control, 2018, Volume 5, 4 issues. ISSN paper version 2325-6818. ISSN online version 2325-6826. Also available as a combined paper and online subscription.
# Contents

1 Introduction
   Background and Motivation ................................................. 2
   Scope of the Monograph .................................................... 4
   Organization .................................................................. 7

1 Theory ...................................................................... 9

2 Preliminaries ................................................................ 10
   Mathematical Background .................................................. 10
   Models for Distributed Control Systems .............................. 20
   Distributed Algorithms ...................................................... 22
   Further Reading ............................................................... 26

3 Distributed Asymptotic Average Consensus ...................... 28
   Introduction and Motivation ............................................. 29
   Preliminaries and Problem Formulation ............................. 31
   Distributed Averaging in Undirected Graphs ..................... 39
   Distributed Averaging in Digraphs ..................................... 45
   Overcoming Transmission Delays and Packet Drops .......... 65
   Privacy and Security Considerations ................................. 72
   Discussion .................................................................. 78
4 Distributed Finite-Time Average Consensus 81
   Introduction and Motivation .......................... 81
   Exact Finite-Time Average Consensus .................. 82
   Approximate Finite-Time Average Consensus .......... 99
   Discussion ............................................. 105

5 Distributed Weight Balancing 108
   Introduction and Motivation .......................... 108
   Preliminaries and Problem Formulation ................. 110
   Distributed Balancing with Unconstrained Weights .... 115
   Distributed Balancing with Constrained Weights ...... 123
   Balancing in Undirected Graphs ......................... 140
   Discussion ............................................. 142

II Applications 145

6 Coordination of Distributed Energy Resources 146
   Introduction and Motivation .......................... 146
   Power Distribution Network Model ....................... 147
   Fair-Splitting Allocation .............................. 150
   Optimal Allocation .................................... 154
   Flow-Feasible Allocation .............................. 158
   Discussion ............................................. 166

7 The PageRank Problem 168
   Introduction and Motivation .......................... 168
   The PageRank Problem .................................. 170
   Centralized Computation of PageRank Values .......... 171
   Asynchronous Computation of PageRank Values ......... 172
   The Asynchronous PageRank Algorithm .................. 178
   Discussion ............................................. 180

References 182
Distributed Averaging and Balancing in Network Systems

Christoforos N. Hadjicostis, Alejandro D. Domínguez-García and Themistoklis Charalambous

1 University of Cyprus, Cyprus
2 University of Illinois, USA
3 Aalto University, Finland

ABSTRACT

Rapid developments in digital system and networking technologies have led to the emergence of complex systems that are de facto managed and controlled over cyber infrastructures, such as wireless and wired broadband networks. The emergence of this type of network systems, which range from smart grids and traffic networks of various sorts, to embedded electronic devices and robotic networks, has sparked huge interest in distributed control problems. This is due to the need to properly coordinate the information exchange between sensors, actuators, and controllers in order to enforce a desirable behavior, without relying on a centralized decision maker. In this monograph, we present some recent progress in this area by focusing on the key operations of distributed average consensus and weight/flow balancing under a variety of communication topologies and adversarial network conditions, e.g., delays, and packet drops. These operations are key in control, coordination, and optimization tasks in many emerging applications; two of these, which we discuss in detail, are the coordination of distributed energy resources, and the computation of PageRank values.

Introduction

Background and Motivation

Over the past few decades, the design of protocols and algorithms for distributed control and coordination in network systems, and more generally distributed function calculation, has attracted significant attention by the control, communication, and computer science communities (see, e.g., Bertsekas and Tsitsiklis (1989), Lynch (1996), Bullo et al. (2009), and Mesbahi and Egerstedt (2010), and the references therein). For example, given a set of interconnected nodes (which could be sensors in a sensor network, routers in a communication network, or unmanned vehicles in a multi-agent system), the nodes may be interested in (i) averaging a set of measurements, (ii) coordinating their speed or direction, (iii) jointly regulating/coordinate traffic in an urban network, (iv) transmitting data from one/multiple sources to one/multiple sinks, or (v) electing a leader with each node casting a vote.

One coordination problem that has been the subject of extensive work by many researchers within the control community since the early 2000s is the so-called consensus (or agreement) problem (see, e.g., the survey by Olfati-Saber et al. (2007) for some of the early work conducted by this community). In this problem, each node in the network initially possesses some value, which is in general different for every node. The objective is then for the nodes to obtain
the value of some function, the argument of which is a vector comprised by the values possessed by the nodes. In general, the nodes may not know what the function is, or they may not have access to all the entries of the argument to be passed to it. It is worth pointing out that while researchers in the control community became interested in the consensus problem in the early 2000s, its roots can be traced back to work in opinion dynamics (see the pioneering works of DeGroot (1974) and Chatterjee and Seneta (1977)), and the seminal work of Tsitsiklis (1984), in the context of distributed computing.

The average consensus problem is a special case of the consensus problem described above, where the nodes’ task is to compute the average of the values they possess initially (see, e.g., Xiao and Boyd (2004)). In this case, nodes know a priori the function to be calculated (i.e., the average), but (as in the consensus problem) they do not necessarily have access to all the entries of the vector-valued argument to be passed to the function. More generally, the nodes might be interested in computing a weighted average of the values they possess, where, perhaps, the weight associated to each value is information only known to the particular node. In this case, despite the fact that nodes may know what the structure of the function is (i.e., a weighted average), they do not necessarily know all function parameters.

While there are many ways to solve consensus problems in a distributed fashion, e.g., flooding (see, e.g., Ho et al. (1999)), a popular approach is to use one (or more) linear iteration, where each node repeatedly updates some variables as a weighted linear combination of the previous values these variables take and those maintained by its neighbors. Each of the iterations in these algorithms can be thought of as an autonomous discrete-time linear system with a (possibly time-varying) transition matrix, also referred to as weight matrix, that is defined by the coefficients (weights) used in the linear updates. Our focus is on discrete-time linear iterations because they allow flexibility to overcome abnormalities that appear in practice (such as communication delays and faulty/malicious components), and are suitable for a variety of emerging systems with hybrid dynamics and event-driven control. We should point out, however, that several of the iterative-type algorithms we describe can be translated to continuous-time formulations.

It is worth mentioning that the idea described above of repeated use of weighted linear combinations for the consensus problem was first described
by Feller (1968) (see the discussion in Example XIII, (10.c), on page 333, and Problem 15 on page 425), who named the procedure \textit{repeated averaging}, and made the connection between the properties of the aforementioned weight matrix and the matrix of a homogenous ergodic Markov chain. This observation was exploited later by DeGroot (1974) for the case of fixed weights. Repeated averaging with time-varying weight matrices was rigorously treated in Chatterjee and Seneta (1977), where the authors made connections to the theory of non-homogeneous Markov chains, and leveraged many results developed in this area. In this regard, the book by Seneta (2006) (originally published in 1973) is a must-read reference as it contains many tools to analyze consensus algorithms that use the idea of repeated averaging, with either time-invariant or time-varying weights.

Depending on the application, the choice of weights when using repeated averaging for consensus problems can be challenging. For example, when weights are time-invariant and the objective is for the nodes to converge to a common value, one needs to choose the weights so that the weight matrix is row stochastic (i.e., the weights on each row add up to one). However, the choice of time-invariant weights needs to be carefully done when the objective is for the nodes to obtain the value of some function. For example, when the nodes are interested in computing the average of the values they initially possess, the weights need to be balanced, i.e., they need to be chosen so that the resulting weight matrix is doubly stochastic (i.e., the weights on each row add up to one \textit{and} the weights on each column add up to one); this can be very challenging if the nodes need to obtain such set of weights in a distributed fashion (see, e.g., Gharesifard and Cortés (2012)). In addition, weight balancing can be critical in several other applications, including flow networks of various sorts (e.g., traffic networks and electric power networks), and adaptation/synchronization in complex networks (see, e.g., DeLellis \textit{et al.} (2010) and Yu \textit{et al.} (2012)).

\textbf{Scope of the Monograph}

The focus of this monograph is to address several issues that arise when attempting to utilize repeated-averaging-type algorithms for distributed averaging and balancing in practical settings. These operations are key in control, coordination, and optimization tasks in many emerging applications, including
modern power distribution systems, traffic networks, embedded electronic devices, and robotic networks.

**Communication Constraints**

The communication constraints imposed by the distributed nature of the system need to be taken into account when designing distributed algorithms, either for averaging or for weight/flow balancing. A number of protocols have been developed to address these two problems in distributed systems in which no component may have the capability to communicate directly with all other components. Chapters 3 and 4 address distributed averaging, and Chapter 5 addresses distributed balancing, for both the cases of bidirectional and non-bidirectional exchange of information among pairs of nodes.

**Imperfect Communications**

Coping with the unreliability of the communication channels connecting pairs of nodes in the network is an important challenge. For example, if a communication link between two nodes fails permanently, the nodes need to be able to detect this issue and compensate for it by adapting their weights or via some other means. Also, unreliable communication links can cause transmissions at certain time steps to be delayed or completely dropped. For example, in wireless networks, each node should generally be able to communicate with its neighbors; however, such transmissions may become unreliable and temporarily lost, due to, for example, channel fading and interference from other sources. Acknowledgements allow senders to know whether their transmissions have been received, but this imposes additional overhead and delay, and might not be as straightforward to implement in the case of non-bidirectional exchange of information among pairs of nodes. Chapter 3 discusses these challenges in more detail, and provides algorithms for distributed averaging that overcome them.

**Weight Choice**

As discussed earlier, in repeated-averaging-type algorithms, each component in the system updates a set of variables using weighted linear combinations of the variables of its in-neighbors, i.e., the components that can send information
to them. Depending on the task to be performed, the weights to be used need to be chosen in a particular way. In this regard, and assuming there are no constraints on the values the weights can take, if the exchange of information among the nodes in the system is bidirectional, the choice of weights is usually relatively straightforward and the nodes can do it in a distributed fashion with fairly minimal information. On the other hand, if the exchange of information among the nodes is non-bidirectional, the problem of choosing the weights is much harder even if there are no constraints on the weight choice, and the computations needed for such choice are performed by a single processor with access to all the information defining the problem. The problem complicates even further if there are restrictions on the values that the weights can take. Chapters 3 and 5 address weight choice under different constraints for different types of network systems.

**Execution and Time Complexity vs Precision**

There are many tradeoffs in distributed algorithms for averaging, including the computational/communication complexity, the execution time of the algorithm, and the precision of the outcome. For example, flooding techniques (see, e.g., Ho et al. (1999)) have the components exchange messages until each component in the system becomes aware of all values that need to be averaged; thus, they enable them to compute the average or any other function of the values after a finite number of iterations (that depends on the size and structure of the network), at the cost of high memory requirements and communication complexity (especially if the size of the network is large). On the other hand, iterative strategies that rely on each node updating its value using information that is available from its immediate neighbors, do not impose high memory requirements, but they require an indefinite number of iterations; they typically converge only asymptotically, implying a compromise in precision if they are aborted after a finite number of iterations. Chapter 3 focuses on asymptotic strategies for average consensus, whereas Chapter 4 focuses on finite-time strategies, including strategies that guarantee an approximation of the average within an *a priori* chosen precision and strategies that guarantee the exact average (at the cost of higher computational complexity).
Interplay between Cyber and Physical Layers

Most literature on distributed averaging is focused on algorithm development and the cyber layer for communication and computation on which the algorithms developed are implemented. Unless the end goal is the computation itself—this is the case in a distributed computing system—this may not be sufficient. In general, the nodes in the cyber layer are controlling some actuators or measuring some quantities in a physical system with the objective of making the physical system behave in some particular way. While taking into account the physical layer is key in understanding the overall system behavior, the physical layer model depends heavily on the particular application, and thus it is difficult to abstract out and generalize. For example, in an electric power distribution network, the nodes in the cyber layer may be controlling the amount of power injected by power generating resources. This in turn will result in power flows across the electrical lines connecting the different nodes in the electrical distribution network; these flows are governed by the physics of the system, i.e., Kirchhoff’s laws. Under certain assumptions on the electrical network topology and the operating conditions, the flow of power in an electrical network can be described by a network-flow-theoretic model (see, e.g., Ford and Fulkerson (2010)), which is also used in transportation networks. Thus, in Chapters 5 and 6, we consider such network-flow-theoretic models and their interaction with the cyber layer controlling them.

Organization

The monograph is divided into two parts; these are Part I: Theory (Chapters 2–5), and Part II: Applications (Chapters 6–7). A reader interested mainly in the averaging problem can focus only on Chapters 2, 3, and 4. A reader interested in weight choice and weight/flow balancing can focus on Chapters 2 and 5. A reader interested only in electric power applications can read Chapters 2, 3, 5, and 6. A reader interested only in the PageRank problem can read Chapters 2, 3, and 7.

We include proofs that are easy to present, do not break the flow of the document, and provide some intuition for the results; however, we omit the more complex proofs, and refer the reader to particular references that can be used to further pursue these results.
Instead of having a “centralized” literature review in this introductory chapter, and in following with the spirit of this monograph, we adopt a “distributed” literature review approach, where each chapter contains a review of the references that are relevant to the particular chapter content.
References


Full text available at: http://dx.doi.org/10.1561/2600000016


