AFRL-AFOSR-VA-TR-2019-0039

Reliable Multi-Agent Control in Failure-Prone Environments via Inhomogeneous Markov Chains

Alex Olshevsky
UNIVERSITY OF ILLINOIS

01/04/2019
Final Report

DISTRIBUTION A: Distribution approved for public release.

Air Force Research Laboratory
AF Office Of Scientific Research (AFOSR)/ RTA2
Arlington, Virginia 22203
Air Force Materiel Command

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*
06-03-2019

2. REPORT TYPE
Final Performance

3. DATES COVERED *(From - To)*
15 Sep 2015 to 14 Mar 2019

4. TITLE AND SUBTITLE
Reliable Multi-Agent Control in Failure-Prone Environments via Inhomogeneous Markov Chains

5a. CONTRACT NUMBER

5b. GRANT NUMBER
FA9550-15-1-0394

5c. PROGRAM ELEMENT NUMBER
61102F

6. AUTHOR(S)
Alex Olshevsky

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
UNIVERSITY OF ILLINOIS
506 S WRIGHT STREET SUITE 364
URBANA, IL 61801-3649 US

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
AF Office of Scientific Research
875 N. Randolph St. Room 3112
Arlington, VA 22203

10. SPONSOR/MONITOR'S ACRONYM(S)
AFRL/AFOSR RTA2

11. SPONSOR/MONITOR'S REPORT NUMBER(S)
AFRL-AFOSR-VA-TR-2019-0039

12. DISTRIBUTION/AVAILABILITY STATEMENT
A DISTRIBUTION UNLIMITED: PB Public Release

13. SUPPLEMENTARY NOTES

14. ABSTRACT
This project developed protocols for the coordination of autonomous nodes (such as UAVs, ROVs, and other robotic platforms) operating in fault-prone environments characterized by noisy and time-varying communication, message losses, and persistent and unpredictable node failures. For problems of formation control, leader-following, cooperative estimation and learning, resource allocation, and others tasks fitting within a separable optimization framework, control strategies were developed with fast and reliable performance, even in simulated networks of thousands of nodes.

15. SUBJECT TERMS
Reliable Multi-Agent Control, Failure-Prone Environments, Inhomogeneous Markov Chains

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			LEVE, FREDERICK
Unclassified	Unclassified	Unclassified	UU		19b. TELEPHONE NUMBER *(Include area code)* 703-696-7309

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

DISTRIBUTION A: Distribution approved for public release.

Reliable Multi-Agent Control in Failure-Prone Environments
Final Report

Alex Olshevsky

December 21, 2018

1 Overview

This project develops control strategies for a future in which multi-agent systems, typically composed of groups of UAVs or other robotic platforms, execute military missions currently performed by humans. While such systems will be guided by humans, this human guidance should be "high level," typically by setting objectives for the systems; all the low-level task planning should be performed autonomously by the agents themselves. The objective of this project is to design methods for doing this well for a wide array of network control objectives.

Our benchmark multi-agent control problems are formation control, leader-following, allocation of a divisible resource such as bandwidth, and cooperative estimation/learning. These were chosen due to their cross-cutting applicability throughout many application domains. However, these applications were treated within a general optimization framework so that the advances within this project are "portable" to problems beyond those considered here.

Over the past decade many researchers in the field of multi-agent control have been developing distributed controllers for accomplishing many diverse network objectives, including those we study here. This proposal builds on that work to focus on a number of directions making network control faster, more resilient, and scalable. Specifically, we are interested in three thrusts. First, can we scale multi-agent control to thousands of agents or even more? Many multi-agent control protocols tend to break down by slowing down as the number of agents grows. Secondly, can we design multi-agent control strategies that are resilient to perturbations and noise, even in the regime when the number of agents is large? Finally, at a general level, multi-agent control strategies solve network-wide optimization problems; can we design strategies that solve such problems in a way that is competitive with the best centralized method? In other words, we want to design distributed controllers that perform as well as a a hypothetical centralized decision maker that has access to all the information in the network one place. The ensuing sections detail progress on each of these thrusts.

2 Scalability

How well does multi-agent control scale *with the number of agents*? One natural metric is the number of "rounds," where each round consists of each agent sensing the position of neighboring agents, exchanging messages with the same agents, and then moving to a new location. How many rounds does it take n agents to come close to an objective across a collection of useful tasks such as formation control, leader following, cooperative learning, or resource allocation?

It is common to consider protocols which, on a network of n agents, take $O(n^2)$ and $O(n^3)$ rounds, as scalable. This intuition comes from computer science, where a round is typically a computation; but in multi-agent control, a round involves a position adjustment. An algorithm which took, for example, n^3

rounds would, in a network of 100 agents, take 100,000 position adjustments, making it quite impractical in most real-world situations.

The mathematical tool used in many multi-agent control is the so-called consensus problem, and in paper [J1] we developed a protocol for the consensus problem which takes $O(n)$ rounds on a network with n agents. *This immediately leads $O(n)$ protocols for formation control, leader-following, as well as cooperative learning as well as general models of distributed optimization.* This is orders of magnitude faster compared to all previous control strategies for multi-agent systems.

We illustrate these results in the figures below. Figure 1 shows formation control on the 2D grid, while Figure 2 shows formation control on the 3D grid. The ensuing Figure 3 and Figure 4 show convergence times for these formations, using our proposed formation control strategies, as a function of the number of nodes. *The linear scalability is easily seen on both graphs.* For example, 200-300 position adjustments suffice for both of these formations even when the number of nodes is in the thousands.

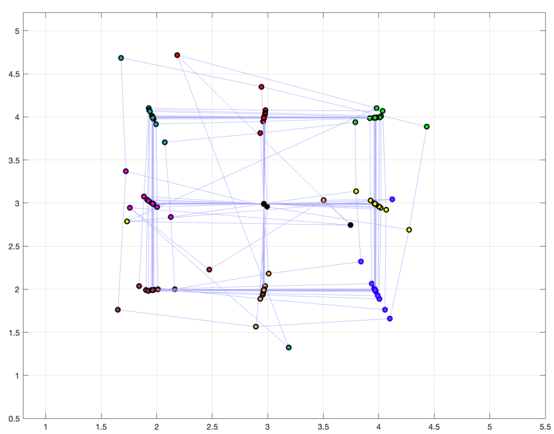

Figure 1: Formation control: 2D grid formation

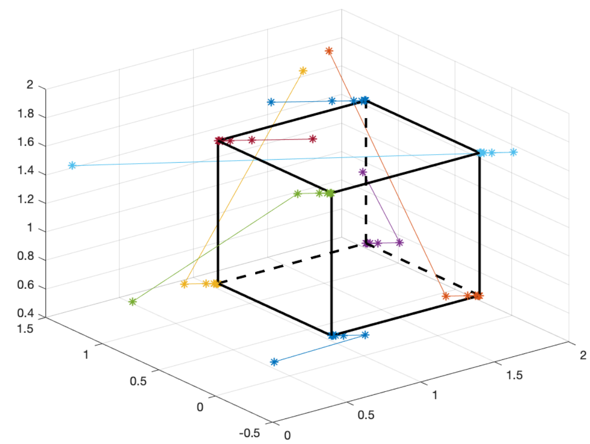

Figure 2: Formation control: 3D grid formation

3 Noise Resilience

A classic line of research within control shows that large interconnected system may be "brittle" in the sense of being destabilized by small perturbations. Examples of this phenomena date to work in the 1960s on so-called "string instability." In recent years, the same phenomenon was observed for multi-agent control systems.

Our main advance on this thrust is a precise understanding of just how resilient a formation is with respect to noise. We consider a formation control protocols defined on a undirected graphs G, in which every node knows what the position of its neighbors should be in its own coordinate system. Formation control can then be done in a distributed way by performing gradient descent on the function $\sum_{i,j} w_{ij}||p_i - p_j - r_{ij}||_2^2$, where p_i is the position of node i while r_{ij} is the desire offset between nodes i and j, and w_{ij} are weights which are nonzero whenever an edge exists in G between i and j. This is the standard model for formation control from offsets – but how resilient is it with respect to noise?

In the paper [J4] we showed that a formation control with independent noise with variance σ^2 at each node will converge to a neighborhood of the optimal formation, with expected square distance of $\sigma^2 K$ to the closest formation. Here K is the so-called Kemeny constant, defined as the expected time to reach a random node in the Markov chain with weights w_{ij}. *The novelty of this result is that it is an equality: we*

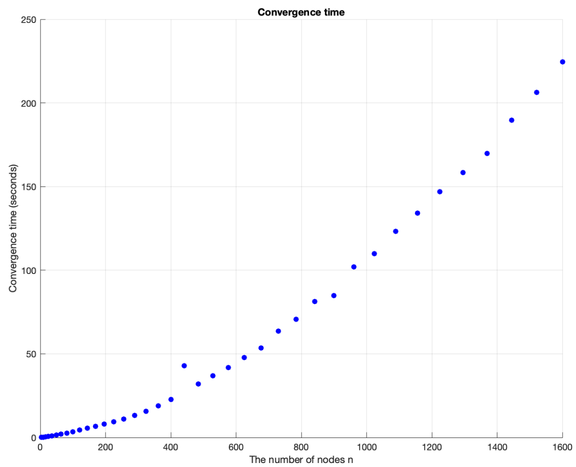

Figure 3: Convergence time: formation control on the 2D grid. The number of nodes is on the x-axis, time to formation is on the y-axis.

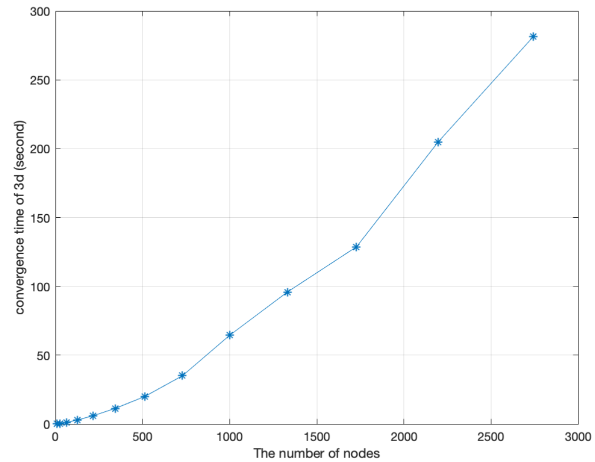

Figure 4: Convergence time: formation control on the 3D grid. The number of nodes is on the x-axis, time to formation is on the y-axis.

exactly characterize the limit of the square distance to the closest formation. All previous work, to the PIs knowledge, had only succeeded in bounding this quantity, in contrast to the exact characterization developed in this project.

This characterization immediately provides guidelines about scalable formations. For example, as discussed in [J4], for a star graph on n nodes we have that $K = O(n)$ while for a binary tree on n nodes we have $K = O(\log n)$. In other words, the binary tree formation is exponentially more resilient to noise than the star formation! We illustrate this finding empirically in the next set of figures. Figures 5 and 6 show formation control on the star graphs; as it can be seen, this is not scalable, and the performance degradation in going from 7 to 127 agents is severe. On the other hand, Figures 7 and 8 show us the same formation control protocol on the complete binary tree, where we see a more graceful degradation, in line with our theoretical predictions.

More broadly, our characterization of resiliency of formation control in terms of the Kemeny constant allows us to make the same calculations for *any* formation. By identifying the key quantity which measures noise resilience, we are able to separate formations which scale well from formations with scale poorly; a table of formations with associated scalings may be found in [J4].

4 Optimality

A lot of multi-agent control boils down to solving an optimization problem: a global set of positions must be agreed upon which need to be optimal for a certain cost/objective, and this agreement can only take place as a consequence of local actions. To that end, we study general distributed protocols for optimization problems. The benchmark problems in this thrust are cooperative estimation and cooperative learning.

Supposing that node i collects a certain amount of data, and consequently knows a function $f_i(w)$ which fits how well the model w fits the data at node i, the nodes want to collectively agree of a minimizer of

$$F(w) = \sum_{i=1}^{n} f_i(w). \tag{1}$$

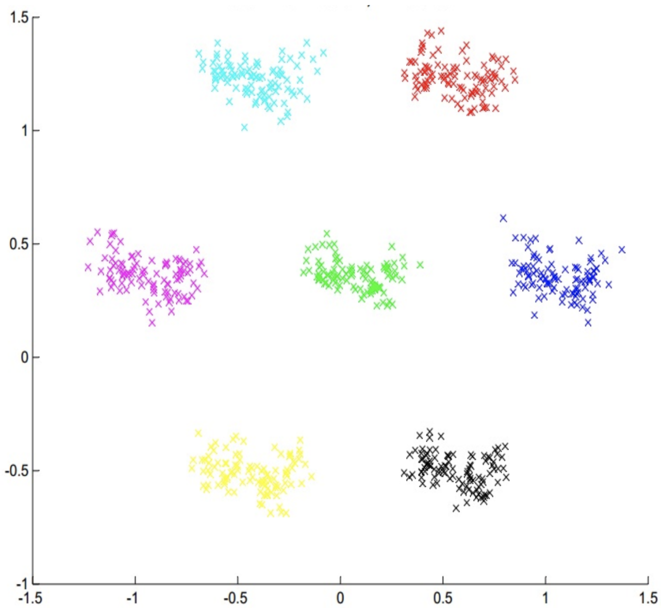

Figure 5: Formation control on the star graph with 7 nodes with noise.

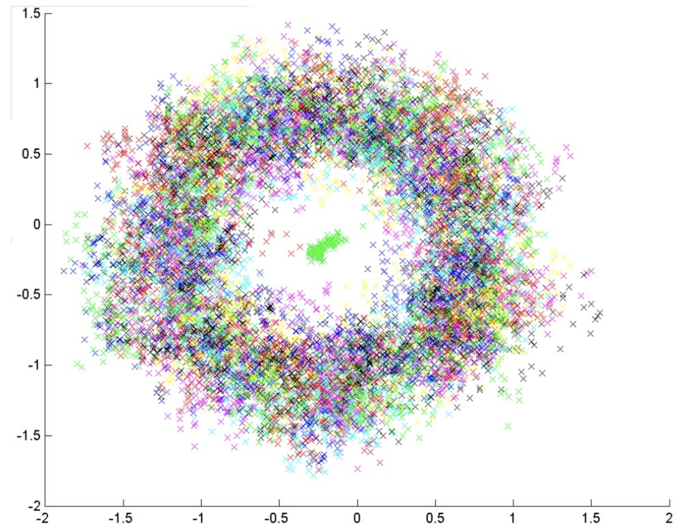

Figure 6: Formation control on the star graph with 127 nodes with noise.

In words, the nodes want to find a parameter w which fits all the data across all the nodes. The catch is that only node i knows its local function $f_i(w)$.

The generality of the model allows it to be used in many application domains. For example, $f_i(w)$ could denote how well a neural network with weights w explains that data collected by node i. Our work in this model focuses on training support vector machines (SMVs), but it could be useful even for much simpler estimation problems.

A sequence of papers produced by the PI analyzed methods to minimize $F(w)$ in a network. In [J2] the first method which converges geometrically over time-varying graph was given. In [J5], a special case of this problem was studied where (the dual of) $F(w)$ represents the best way to allocate a divisible resource, such as bandwidth, across a network. The conference papers [C2] and [C5] represented conference versions of this work. The final outcome of this line of research was the development of an *asymptotically network independent* method in the paper [J6], which we describe next.

To explain the significance of our advances, we begin by observing that, to minimize Eq. (1) over a network of n agents results in two competing effects. On the one hand, a host of real-world problems (e.g., node crashes, message losses, delays, asynchrony, time-varying networks) will create less-than-perfect coordination among nodes. On the other hand, the total computing power has been multiplied by a factor n, at least relative to a hypothetical single processor that has access to all the data in one place. Obviously the first effective is negative while the second effect is positive. In all past research on the topic, the first effect has outweighed the second: distributed optimization methods were worse than their centralized counterparts. In his past work, the PI has often referred to this as the *price of decentralization*.

As a result of this project, this is no longer the case. In particular, the paper [J6] developed an optimization method for Eq. (1) which, after a transient period, exhibit the *same performance* as a centralized method with the same total processing power as all the nodes in the network combined. This has been accomplished under the assumption that the functions $f_i(w)$ are strongly convex with Lipschitz gradients. It is this property that we refer to as asymptotic network independence, meaning that, after a transient, the network does not matter.

We illustrate these results in Figure 9 and Figure 10 for the problem of distributed training of an SVM.

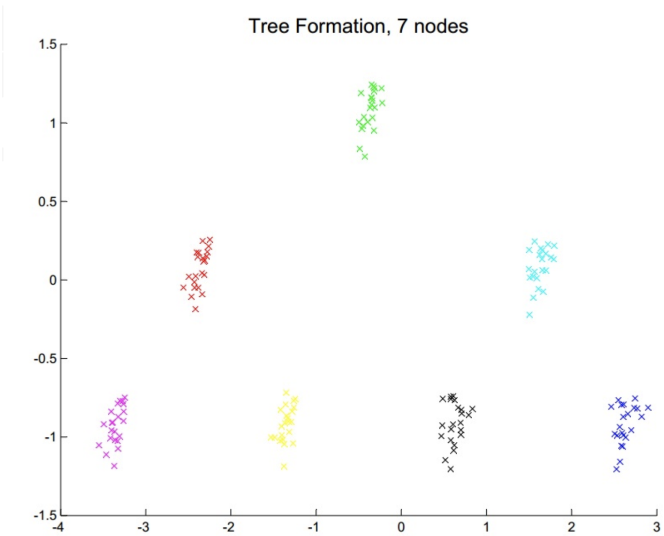

Figure 7: Formation control on the tree graph with 7 nodes with noise.

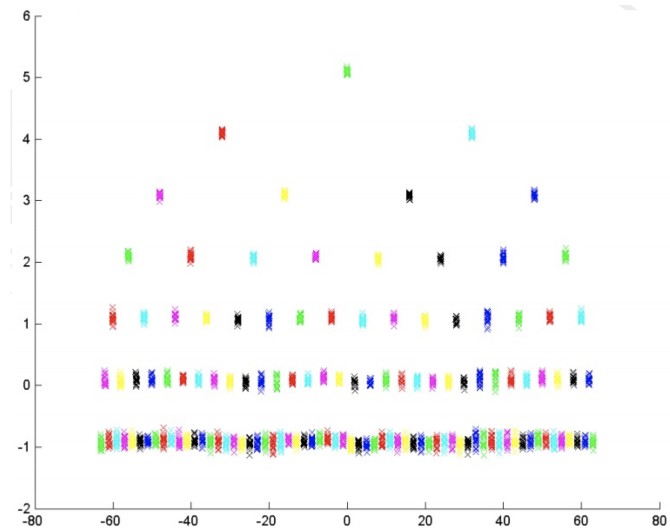

Figure 8: Formation control on the tree graph with 127 nodes with noise.

The classification problem itself is showed in Figure 9; the goal is to find a binary linear classifier to separate the orange and blue points in the regime when there is a small but non-negligible amount of overlap. This problem can easily be solved using a variety of centralized methods. The key question for us is how well it can be solved with *distributed* methods.

Our asymptotically network independent method is simulated in Figure 10. The orange line in Figure 10 represents the performance of the best centralized training procedure – that is, of a hypothetical centralized processor that has access to all the functions $f_1(w), \ldots, f_n(w)$. The dark blue line represents the performance of the distributed methods. As can be seen, during an initial transient period the dark blue line is far above the orange line; but, over time, the dark-blue line catches up, and *in the end the distributed and best possible centralized are indistinguishable.* Finally, the light blue line represents disagreement among the various nodes (demonstrating that, in fact, several distinct classifiers are found by different nodes of the system).

5 Outreach

The PI wrote three tutorials as a part of this project:

- The paper [J3] was provided new and simple proofs of basic facts in distributed optimization. The paper was a self-contained introduction to the field. Published in the *Proceedings of the IEEE*, the paper was written with the goal of making the field accessible to a wider array of researchers.
- The papers [C3] and [C4] accompanied tutorials given at the *IEEE Conference on Decision and Control*. While [C4] explained how much of multi-agent control can be done in linear time (see Section 2 for details), [C3] focus on one application area – distributed learning – and explained the repercussion of the results in Section 4 for that application domain. The goal of both papers was to popularize the outcomes of research done within the scope of this project.

6 Papers produced by this project

Papers are listed in inverse chronological order. All authors are always in alphabetical order.

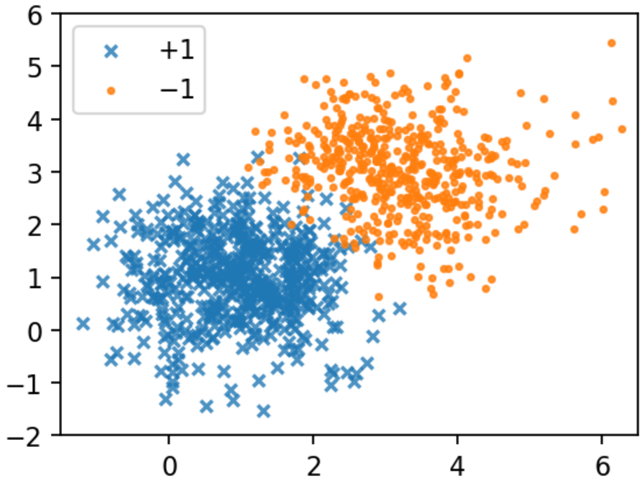

Figure 9: A sample classification problem.

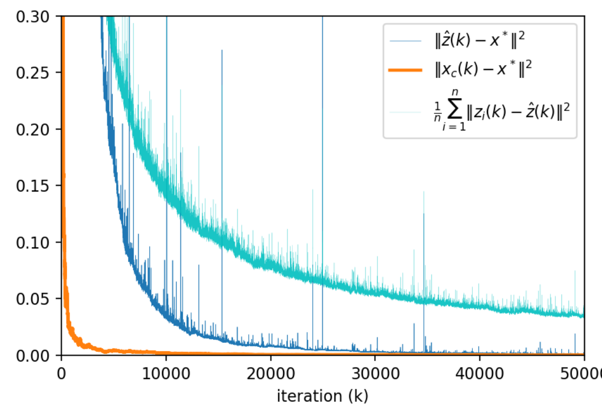

Figure 10: Decay of the squared error for centralized and distributed gradient descent for training an SVM.

6.1 Journal papers

J6. "Robust Asynchronous Stochastic Gradient-Push: Asymptotically Optimal and Network-Independent Performance for Strongly Convex Functions," Alex Olshevsky, Ioannis Ch. Paschalidis, Artin Spiridonoff, submitted to **Journal of Machine Learning Research**.
https://arxiv.org/abs/1811.03982

J5. "Improved Convergence Rates for Distributed Resource Allocation," A. Nedic, A. Olshevsky, W. Shi, submitted to **Computational Optimization and Applications**.
https://arxiv.org/abs/1706.05441

J4. "On Performance of Consensus Protocols Subject to Noise: Role of Hitting Times and Network Structure," A. Jadbabaie, A. Olshevsky, accepted to the **IEEE Transactions on Automatic Control**.
http://arxiv.org/abs/1508.00036

J3. "Network Topology and Communication-Computation Tradeoffs in Distributed Optimization," A. Nedic, A. Olshevsky, M. Rabbat, **Proceedings of the IEEE**, vol. 106, no. 5, pp. 1-24, 2018.
https://arxiv.org/abs/1709.08765

J2. "Achieving Geometric Convergence for Distributed Optimization over Time Varying Graphs," A. Nedic, A. Olshevsky, W. Shi, **SIAM Journal on Optimization**, vol. 27, no. 4, pp. 2597-2633, 2017.
https://arxiv.org/abs/1607.03218

J1. "Linear Time Average Consensus and Distributed Optimization on Fixed Graphs," A. Olshevsky, **SIAM Journal on Control and Optimization**, vol. 55, no. 6, pp. 3990-4014, 2017.
https://arxiv.org/abs/1411.4186

6.2 Conference Papers

C5. "Geometrically convergent distributed optimization with uncoordinated step-sizes," A. Nedic, A. Olshevsky, W. Shi, C. Uribe, **Proceedings of the American Control Conference**, 2017.

C4. “Fast Algorithms for Distributed Optimization and Hypothesis Testing: A Tutorial,” A. Olshevsky, **Proc. of the IEEE Conference on Decision and Control**, 2016.

C3. “A Tutorial on Distributed (Non-Bayesian) Learning: Problem, Algorithm, and Results,” A. Nedic, A. Olshevsky, C. Uribe, **Proceedings of the IEEE Conference on Decision and Control**, 2016.

C2. “A Geometrically Convergent Method for Distributed Optimization over Time-Varying Graphs,” A. Nedic, A. Olshevsky, W. Shi, **Proceedings of the IEEE Conference on Decision and Control**, 2016.

C1. “On Performance of Consensus Protocols Subject to Noise: Role of Hitting Times and Network Structure,” A. Jadbabaie, A. Olshevsky, **Proceedings of the IEEE Conference on Decision and Control**, 2016.