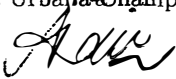

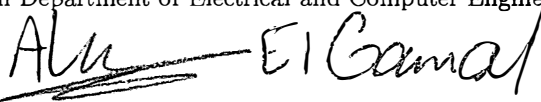
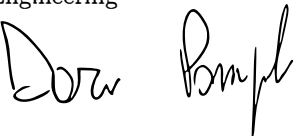


## Defending Large-Scale Distributed Machine Learning Against Adversarial Attacks

**Project Type:** Multi-institution/multi-disciplinary project  
**Total Funds Requested:** \$ 6,000

### Student Co-PIs:

1. Lili Su, lilisu3@illinois.edu (Advisor: Prof. Nitin H. Vaidya)  
Department of Electrical and Computer Engineering  
University of Illinois at Urbana-Champaign;  
Advisor's signature: 
2. Qiong Hu, qionghu.cs@rutgers.edu (Advisor: Prof. Tomasz Imielinski)  
Department of Computer Science  
Rutgers University;  
Advisor's signature:
3. Seyyed A. Fatemi, Sfatemi@hawaii.edu (Advisor: Anthony Kuh)  
Department of Electrical Engineering  
University of Hawaii at Manoa;  
Advisor's signature: 
4. Rehana Mahfuz, rmahfuz@purdue.edu (Advisor: Prof. Aly El Gamal)  
Undergraduate Student in Department of Electrical and Computer Engineering  
Purdue University;  
Advisor's signature: 
5. Vidyasagar Sadhu, vidyasagar.sadhu@rutgers.edu (Advisor: Prof. Dario Pompili)  
Department of Computer Engineering  
Rutgers University;  
Advisor's signature: 

# Defending Large-Scale Distributed Machine Learning Against Adversarial Attacks

**Project Type:** Multi-institution/multi-disciplinary project

**Total Funds Requested:** \$ 6,000

## **Student Co-PIs:**

1. Lili Su, lilisu3@illinois.edu (Advisor: Prof. Nitin H. Vaidya)  
Department of Electrical and Computer Engineering  
University of Illinois at Urbana-Champaign;  
**Advisor's signature:**
2. Seyyed A. Fatemi, Sfatemi@hawaii.edu (Advisor: Anthony Kuh)  
Department of Electrical Engineering  
University of Hawaii at Manoa;  
**Advisor's signature:**
3. Rehana Mahfuz, rmahfuz@purdue.edu (Advisor: Prof. Aly El Gamal)  
Undergraduate Student in Department of Electrical and Computer Engineering  
Purdue University;  
**Advisor's signature:**
4. Vidyasagar Sadhu, vidyasagar.sadhu@rutgers.edu (Advisor: Prof. Dario Pompili)  
Department of Computer Engineering  
Rutgers University;  
**Advisor's signature:**

## 1 Problem Statement

Machine learning, as one of the primary analysis tools in data science, has been attracting intensive attention from researchers [10] and has been adopted and applied widely in the fields as diverse as personal healthcare, manufacturing, financial security, and marketing [3,2,6,12,4]. In literature, much work focuses on problems of the form

$$\min_{\mathbf{w} \in \mathbb{R}^d} f(\mathbf{w}) \triangleq \sum_{k=1}^n \ell(\mathbf{w}; \mathbf{x}_k; y_k) + \lambda R(\mathbf{w}), \quad (1)$$

where  $n$  is the total number of examples,  $\mathbf{x}_k$  and  $y_k$  are the feature vector and the label, respectively, of the  $k$ -th example,  $\mathbf{w}$  is a predictor candidate,  $\ell$  is a loss function,  $\lambda$  is a constant, and  $R$  is a regularizer that imposes desired structural properties of the predictor  $\mathbf{w}$ . This project focuses on the machine learning problems that can be captured by (1).

Big Data necessitates large-scale distributed machine learning for the following reasons. (A) Both the data complexity and the dataset size are exploding due to the rapid growth in the ability of networked computing systems (such as mobile devices, software logs, cameras, RFID readers and wireless sensor networks [15]) to collect a vast amount of high dimensional data. When the data is “Big”, both the number of examples  $n$  is large and the dimension of the feature vector  $\mathbf{x}_k$  is high – making it costly or even impossible to store and process all the data using a single machine [11,8].<sup>1</sup> (B) Individual data owners tend to be connected to a network in order to improve the performance of the learning tasks. In fact, meta machine learning, where individual machine learning systems are connected via communication networks, is one of the emerging trends in machine learning [9]. Due to privacy issues as well as the high communication complexity, communicating the locally collected data to a centralized processing unit is prohibitive. On the other hand, in large-scale distributed machine learning systems, machines are more vulnerable to adversarial attacks (such as hacking), and some individual data owners may not be cooperative – possibly due to the misalignment of interests with others. Despite the significant importance of the fault-tolerance aspect in real-world applications, to the best of our knowledge, few effort has been made on this [14].

Many distributed optimization algorithms and systems have been proposed to solve (1). Those methods exploit its natural decomposability over examples [5,7,1]. For instance, let  $\{(\mathbf{x}_k, y_k)\}_{k=1}^{n_j}$  (for  $j = 1, \dots, m$ ) be the partition of the  $n$  examples in (1), with  $\{(\mathbf{x}_k, y_k)\}_{k=1}^{n_j}$  being the subset of data assigned to or collected by machine  $j$ . Define  $f_j(\mathbf{w}) \triangleq \sum_{k=1}^{n_j} \ell(\mathbf{w}; \mathbf{x}_k; y_k) + \frac{\lambda}{m} R(\mathbf{w})$  as the local objective associated with the locally stored data subset. The goal of the system designer is to, ideally, have the  $m$  machines collectively minimize

$$\sum_{j=1}^m f_j(\mathbf{w}) = f(\mathbf{w}) = \sum_{k=1}^n \ell(\mathbf{w}; \mathbf{x}_k; y_k) + \lambda R(\mathbf{w}), \quad (2)$$

either with the help of additional administration/scheduling machines (parallel computing) or in a fully distributed manner (distributed computing). Most of the existing work assumes that every machine/agent/computing unit is cooperative in the sense that it follows the protocols/rules specified by the algorithm/system designers although it may be slow or crash unexpectedly once in a while [5,7,1,13]. While these assumptions often accurately capture the daily operations of the current distributed (parallel) machine learning systems, when some unknown subset of machines are hacked by an adversary, their behavior can be detrimental to the correct operation of the system. What’s worse, as noted above, in meta distributed systems, individual owners may not be cooperative.

---

<sup>1</sup> Note that subsampling may not suffice when the dimension of the feature vector  $\mathbf{x}_k$  is high.

To provide a focus, this project considers the attack model wherein the adversary agent tries to control the system’s output (learned predictor) that best fits its own interest by hacking computing machines. Our goal is to design algorithms as well as systems that will defend the large scale distributed machine learning against the above adversarial attacks.

## 2 Proposed Activity

In this section, we will first give an overview of the aspects of the problem that we want to explore in this project. Then, as a first step approaching the goal of this project, we will specify our initial problem formulation. At the end of this section, we will list our proposed schedule for this project.

**Overview:** The architecture of a system imposes significant constraints on the scalability of the system, the robustness of the system against adversarial attacks, as well as the communication cost. We will consider two architectures in this project: fully distributed architecture and hierarchical architecture. In the former, the system only consists of computing agents, in contrast to the latter, where master agents (administration agents) exist. We will mainly focus on the fully distributed architecture due to the fact that it is better-suited for Big Data. We are also interested in the impacts of the following key aspects on the robustness of a system against adversarial attacks: (1) network topologies, (2) communication mechanisms, and the (3) memory constraints.

**Initial Problem Formulation:** We will first focus on the following initial problem formulation. We will enrich our problem formulation to better capture the practical requirements as we progress.

Consider  $m$  agents/machines are connected by a network  $G(\mathcal{V}, \mathcal{E})$ , where  $\mathcal{V} = \{1, \dots, m\}$  is the collection of agents/machines and  $\mathcal{E}$  is the collection of communication links. The networked agents exchange messages locally with their neighbors. Each agent  $j$  has a local objective function  $f_j(\mathbf{w})$ , initially, it is known **only** to agent  $j$ . One agent is hacked by the system adversary, with local objective replaced by  $\tilde{f}(\mathbf{w})$ , which is the cost function of the system adversary. In addition, we assume that the adversary knows the protocol that the hacked agent is supposed to follow, but does not know the local objectives of other agents. That is, the adversary only has the same local view of the system as the hacked agent. In a given execution, without loss of generality, we assume that agent  $\tilde{j}$  has been hacked. We refer to the agents that have not been hacked, i.e.,  $\mathcal{V} - \{\tilde{j}\}$ , as cooperative agents; and refer to agent  $\tilde{j}$  as selfish agent (for reason to be clearer soon). Note that initially no cooperative agents know agent  $\tilde{j}$  has been hacked. In addition, the system adversary can choose different agents to hack in different executions.

The goal of the cooperative agents is to cooperatively identify a predictor  $\mathbf{w}$

$$\min_{\mathbf{w} \in \mathbb{R}^d} \frac{1}{m-1} \sum_{j \in \mathcal{V} - \{\tilde{j}\}} f_j(\mathbf{w}), \quad (3)$$

to best predict with respect to all the data collected by the cooperative agents. In contrast, the goal of the system adversary is to control the common predictor of the cooperative agents to

$$\min_{\mathbf{w} \in \mathbb{R}^d} \tilde{f}(\mathbf{w}), \quad (4)$$

the best predictor from the prospective of the adversary. We want to design algorithms to enable the cooperative agents achieve their task described by (3).

**Project Schedule:** Towards this, we plan to undertake the following over the next year:

- Bi-weekly virtual meeting: every other Wednesday evening 7 pm EST for the summer 2016;
- Background reading: For the first one to two months, read relevant literature, such as distributed optimization methods, and distributed machine learning system;
- Brainstorm ideas about solving our first step problem stated in the above paragraph.
- Consider the concrete objective functions which are loss aggregations of local data, and adapt our results for the initial problem formulation to this concrete setting;
- Meet in person at Purdue University to run large-scale simulation of our algorithms;
- Consider the more general problem formulations if the project progresses smoothly.

We envision our efforts to be in line with Big Data, which is one the major driver for large-scale distributed machine learning that we focus on. This is specifically connected with Networks and AI/Learning Statistics.

### 3 Goal(s), and Outcomes:

Our eventual goal is to design algorithms as well as systems that will defend the large scale distributed machine learning against adversarial attacks.

In the short term, we expect to: (1) have every team member have a solid background in distributed machine learning; (2) solve (or show impossibility of) our first step problem by detecting the selfish agent under the simplified assumption that the local objectives kept by the cooperative agents are identical; (3) design algorithms for general local objectives by examining the “similarity” of the local objectives; (4) consider the concrete objective functions which are loss aggregation of local data; (5) meet in person at Purdue to run large-scale simulation of our algorithms.

In the long term (a year) we expect to be able to: (1) solve the general problem formulation briefly discussed in Section 2, and (2) submit a paper at one top machine learning or distributed computing conferences (such as COLT, ICML, NIPS, PODC, DISC), with extended version submitted to the top journals in machine learning or distributed computing (such as JMLR, DIST).

### 4 Proposed work statement:

Each team member will have a unique contribution to this project.

- **Lili Su:** Her research focus lies at the intersection of distributed computing, machine learning, optimization and communications. She is interested in designing distributed algorithms that are robust to adversarial attacks.
- **Seyyed A. Fatemi:** His research focus is on application of signal processing methods in smart power grid specifically prediction of renewable energy resources and operation and control of electric power grid.
- **Rehana Mahfuz:** Her research interest lies in the areas of Data Mining and Machine Learning, with an emphasis on sports applications.
- **Vidyasagar Sadhu:** His research is in to mobile phone sensing, crowdsensing and mobile distributed computing. He is interested in developing innovative artificial intelligence and distributed machine learning/statistical techniques to tackle challenges in these areas.

Detailed project schedule can be found in Section 2. From our positive collaboration experience during the *Multidisciplinary Research and Data Science Workshop*, we believe that our highly self-motivated team members will be productive on the proposed project.

## Budget & Justification Section

As noted in Section 2 and Section 3, we plan to meet in person to run large-scale simulation, and to present our results at the leading conferences in machine learning or distributed computing communities. A rough estimate of expenses associated the proposed activities are summarized as follows:

1. Four team members attend and co-present project results (oral or poster) at appropriate conference = \$4,000
  - ~\$200 per student registration
  - ~\$400 average flight
  - ~\$400 lodging
2. In person meeting at Purdue to run large-scale simulations of algorithms = \$2,000 (3 nights, 2.5 working days at Purdue)
  - Lili (driving \$ 50, on campus hotel \$100/night x 3 nights) = \$350
  - Seyyed (flight \$700 + hotel \$300) = \$1000
  - Vidyasagar (flight \$350 + hotel \$300) = \$650
  - Rehana (already at Purdue)

## References

1. A. Agarwal, O. Chapelle, M. Dudík, and J. Langford. A reliable effective terascale linear learning system. *The Journal of Machine Learning Research*, 15(1):1111–1133, 2014.
2. R. Burbidge, M. Trotter, B. Buxton, and S. Holden. Drug design by machine learning: support vector machines for pharmaceutical data analysis. *Computers & Chemistry*, 26(1):5–14, 2001.
3. P. Chowriappa, S. Dua, and Y. Todorov. *Introduction to machine learning in healthcare informatics*, pages 1–23. Springer Berlin Heidelberg, Berlin, Heidelberg, 2014.
4. G. Cui, M. L. Wong, and H.-K. Lui. Machine learning for direct marketing response models: Bayesian networks with evolutionary programming. *Management Science*, 52(4):597–612, 4 2006.
5. O. Delalleau and Y. Bengio. Parallel stochastic gradient descent. *CIAR Summer School, Toronto*, 2007.
6. S. Fernández, R. Aler, and D. Borrajo. Machine Learning in Hybrid Hierarchical and Partial-Order Planners for Manufacturing Domains. *Applied Artificial Intelligence: An International Journal*, 19(8):783–809, 2005.
7. P. A. Forero, A. Cano, and G. B. Giannakis. Consensus-based distributed support vector machines. *The Journal of Machine Learning Research*, 11:1663–1707, 2010.
8. Z. Huang and A. Gelman. Sampling for bayesian computation with large datasets.
9. M. Jordan and T. Mitchell. Machine learning: Trends, perspectives, and prospects. *Science*, 349(6245):255–260, 2015.
10. D. Michie. Methodologies from machine learning in data analysis and software. *The Computer Journal*, pages 559–565, 1991.
11. S. L. Scott, A. W. Blocker, and F. V. Bonassi. Bayes and big data: The consensus monte carlo algorithm. *International Journal of Management Science and Engineering Management*, 2016.
12. P. Seemakurthi, S. Zhang, and Y. Qi. Detection of fraudulent financial reports with machine learning techniques. In *2015 Systems and Information Engineering Design Symposium*, pages 358–361. IEEE, 4 2015.
13. M. Stefano, M. Vincenzo, and T. Lang. Distributed detection in the presence of Byzantine attack in large wireless sensor networks. *Proceedings of the 2006 IEEE conference on Military communications*, pages 1154–1157, 2005.
14. L. Su and N. H. Vaidya. Multi-agent optimization in the presence of byzantine adversaries: Fundamental limits. In *Proceedings of IEEE American Control Conference (ACC)*, July, 2016.
15. X.-Y. Xiang-Yang Li, Y. Yajun Wang, and Y. Yu Wang. Complexity of data collection, aggregation, and selection for wireless sensor networks. *IEEE Transactions on Computers*, 60(3):386–399, 3 2011.