Automatica 78 (2017) 88–96

Contents lists available at ScienceDirect

Automatica

journal homepage: www.elsevier.com/locate/automatica

Brief paper

# Convergence analysis of Iterated Best Response for a trusted computation game☆

Shaunak D. Bopardikar [a], Alberto Speranzon [a], Cédric Langbort [b]

[a] United Technologies Research Center, 411 Silver Lane, East Hartford, CT 06118, USA
[b] University of Illinois Urbana Champaign, United States

ARTICLE INFO

*Article history:*
Received 31 December 2014
Received in revised form
18 August 2015
Accepted 14 November 2016
Available online 24 January 2017

*Keywords:*
Game theory
Computational methods
Adversarial machine learning
Iterated Best Response

ABSTRACT

We introduce a game of trusted computation in which a sensor equipped with limited computing power leverages a central computer to evaluate a specified function over a large dataset, collected over time. We assume that the central computer can be under attack and we propose a strategy where the sensor retains a limited amount of the data to counteract the effect of attack. We formulate the problem as a two player game in which the sensor (defender) chooses an optimal fusion strategy using both the non-trusted output from the central computer and locally stored trusted data. The attacker seeks to compromise the computation by influencing the fused value through malicious manipulation of the data stored on the central computer. We first characterize all Nash equilibria of this game, which turn out to be dependent on parameters known to both players. Next we adopt an Iterated Best Response (IBR) scheme in which, at each iteration, the central computer reveals its output to the sensor, who then computes its best response based on a linear combination of its private local estimate and the untrusted third-party output. We characterize necessary and sufficient conditions for convergence of the IBR along with numerical results which show that the convergence conditions are relatively tight.



## 1. Introduction

The Internet of Things (IoT) (Internet of Things, 2015) is the next generation internet where many embedded devices (sensors, actuators, controllers, etc.) are interconnected and can exchange data. Although such devices are becoming increasingly more advanced and capable, the amount of data they can process is still a small fraction of what they can collect. As a consequence, IoT devices may need to leverage intermediate but more capable devices that can store and compute over larger data streams.

We study the problem where a sensor (short for IoT device) exchanges data with a larger and more powerful computer to carry out a specific computation on the data the sensor has collected over a certain period of time. Ideally, the sensor would be able to send/stream the data to the computer, which would then compute the function and send back the result. However, we assume here that the data stored in the central computer can be manipulated maliciously by an attacker. In this case, the sensor has three options: (1) to retain a small amount of the data and compute the function of interest on such trustworthy data knowing that the computation will not be as accurate or, (2) take the risk that the attack is only mildly compromising the data on the central computer so that the result of the computation is close to the true value; or, (3) try to exchange partial results iteratively with the central computer and fuse locally trusted computation on small sample with tampered computation on the full dataset.

This paper formalizes the third scenario, which has the other two as limiting cases. In particular, we model this problem of trusted computation as a game between the sensor/central computer and the attacker. We design and analyze a simple protocol in which each player plays its best response and we study its convergence properties. Additionally, this paper makes a "worst case" assumption, namely that the attacker knows exactly the fusion strategy adopted by the sensor.

The approach we consider in this paper is related to an emerging field called *adversarial machine learning*, wherein two parties,

☆ This work was funded by United Technologies Research Center (UTRC) under the Cyber-Physical Security Initiative. Alberto Speranzon was with UTRC at the time this work was carried out. He is now with Honeywell Aerospace. The material in this paper was partially presented at the 2015 American Control Conference, July 1–3, 2015, Chicago, IL, USA (Bopardikar et al., 2014). This paper was recommended for publication in revised form by Associate Editor Hideaki Ishii under the direction of Editor Ian R. Petersen.

*E-mail addresses:* bopardsd@utrc.utc.com (S.D. Bopardikar), alberto.speranzon@honeywell.com (A. Speranzon), langbort@illinois.edu (C. Langbort).

http://dx.doi.org/10.1016/j.automatica.2016.11.046
0005-1098/

a learner and an attacker, are involved, see Biggio, Nelson, and Laskov (2012), Huang, Joseph, Nelson, Rubinstein, and Tygar (2011) and Sawade, Scheffer, Brückner, and Schffer (2013). The learner uses data to train, for example, a classifier or a regressor, and the attacker is modifying the data so that the learner ends up training the algorithm incorrectly. In this context, the problem is posed as a Bayesian game (Sawade et al., 2013), where the learner minimizes the effect of the attack on the learning algorithm, whereas the attacker maximizes the deviation of such learning algorithm from the correct result and towards a strategically chosen outcome, under the assumption that only a subset of the data can be modified. In Sawade et al. (2013), for example, the e-mail spam problem is considered, where the learner is set to train a classifier to discriminate between spam and non-spam, while the attacker tries to maximize the chances that a spam is classified as non-spam.

The approach considered in this paper also relates to the procedure of *fictitious play (FP)*. In this procedure, each player tries to learn the probability distribution from which the opponent is drawing its actions (Brown, 1951; Robinson, 1951). A recent body of work in the control literature analyzes convergence of fictitious play for several scenarios (Marden, Arslan, & Shamma, 2009; Shamma & Arslan, 2004, 2005). In particular, Shamma and Arslan (2004) present unified energy-based convergence proofs that work for several special classes of games under FP. In Shamma and Arslan (2005), convergence to Nash equilibria is analyzed under the assumption that each player can access the derivatives of the update mechanisms, leading to dynamic FP. A variant of FP, known as Joint Strategy FP, is proposed and the convergence analyzed for several classes of games, especially in high-dimensional spaces, see Marden et al. (2009). More recently, Gaussian cheap talk games, such as Farokhi, Texeira, and Langbort (2014), have been considered. In this context, a sender (adversary) sends corrupted information to a receiver (sensor) under the assumption that the adversary has full knowledge of the receiver's private information.

### 1.1. Main contributions

The contributions of this paper are four-fold. First, we formulate a new problem on trusted computation within a game-theoretic framework and adopt an Iterated Best Response (IBR) (Fudenberg, 1998; Reeves & Wellman, 2004) algorithm to compute final strategies for the sensor and the attacker. More specifically, we consider a protocol such that at each iteration, the attacker reveals its output to the sensor that then computes its best response as a linear combination of its private local estimate and of the untrusted output. The attacker can then, based on the announced policy of the sensor, decide its best response. There is a clear mismatch in the information pattern between attacker and sensor and, in particular, the fact that the attacker cannot access the realization of the private local estimate of the sensor distinguishes this work from the information pattern considered in cheap talk games (Farokhi et al., 2014).

Second, we characterize conditions on the existence of equilibria of the game. These conditions and the equilibria themselves turn out to be functions of all of the problem parameters, viz., the private information belonging to both players. Therefore, to obtain results from a single player's perspective, a third contribution of this paper is to define two notions of convergence for the IBR algorithm, depending upon whether the algorithm converges for some initial value picked by the attacker (*weak convergence*), or for every initial value (*strong convergence*). We derive necessary conditions for weak convergence and sufficient conditions for strong convergence. If the algorithm converges, then it also tells the sensor how to optimally fuse its private estimate with the output. We identify regimes in which some sufficient conditions are also necessary. Numerical simulations indicate that the conditions are relatively tight.

Fourth and finally, the analysis in this paper allows for a certain level of *mismatch* in the distributions used by the players in computing their respective cost functions. This generalizes the analysis presented in the preliminary conference version (Bopardikar, Speranzon, & Langbort, 2014), which assumed that the attacker knows precisely the mean of the distribution used by the sensor to compute its private estimate.

Given that the proposed framework requires an iterative process between sensor and the central computer, the algorithm presented in this paper could be suitable for computation algorithms that are iterative in nature so that partial results can be exchanged between sensor and the central computer. Examples are eigenvalue/eigenvector computation, matrix factorization, iterative optimization methods, etc. The connection of this work to control theory lies in the fact that convergence analysis of the IBR essentially leads to a closed loop dynamical system. This aspect is similar in flavor to the set-ups analyzed in Marden et al. (2009) and Shamma and Arslan (2004, 2005). Using geometric relationships to bound the evolution, we determine necessary and sufficient conditions on the parameters involved which will lead to stability from any/some initial conditions.

### 1.2. Paper organization

The paper is organized as follows. The problem formulation and the proposed approach is described in Section 2. Conditions for the existence of equilibria together with an insightful geometric interpretation are presented in Section 3. Convergence results for the IBR algorithm are derived in Section 4 along with supporting numerical results. Finally, conclusions and directions for future research are discussed in Section 5. Proofs of the results in this paper as well as improved results for the case of equal means from Bopardikar et al. (2014) are available online at Bopardikar, Speranzon, and Langbort (2016).

## 2. Problem formulation

The problem scenario is depicted in Fig. 1. This work assumes that the data and the computation to be carried out are such that it is possible for the sensor to compute an estimate of the true output locally using some random subset of the data, and that the statistics (up to the mean with a finite second moment) about how the actual value is distributed given a value of estimate is known to the sensor. For example, suppose that the data $d \in \mathbb{R}^{N\times M}$, consisting of $N$ data points each represented by an $M$-dimensional feature vector, is uploaded through a trusted sensor to the third-party computer. During the upload process, the sensor could retain a randomly sparsified sample $\hat{d} \in \mathbb{R}^{N\times M}$ of the original data $d$. The data, once stored on the central computer, can be compromised by the attacker leading to a different value $\bar{d} \in \mathbb{R}^{N\times M}$. The sensor seeks to compute the true value of the function $y = g(d) \in \mathbb{R}^k$ of the data $d$ where $g : \mathbb{R}^{N\times M} \to \mathbb{R}^k$ is an algorithm of interest. For example, $g(d) \in \mathbb{R}$ could be the maximum singular value of a matrix with $\hat{d}$ being a sparsified sample of $d$. The sensor has only an approximate knowledge, $\hat{y} = g(\hat{d}) \in \mathbb{R}^k$, of $y$, obtained from the sparsified data $\hat{d}$. The third-party computer computes the value $\bar{y} = g(\bar{d}) \in \mathbb{R}^k$ based on the corrupted data $\bar{d}$. This paper does not address the problem of how to construct a sparsified sample $\hat{d}$, but rather makes an implicit assumption that for a given number of non-zero elements in $\hat{d}$ and a corresponding sparsification procedure, the distribution of $y$ given $\hat{y}$ can be characterized a priori. The actual construction of a specific sampling procedure for a computation such as the maximum eigenvalue would be a topic of future