Abstract

This paper aims at secure and privacy preserving consensus algorithms of networked systems. Due to the technical challenges behind decentralized design of such algorithms, the existing results are mainly restricted to a network of systems with simplest first-order dynamics. Like many other control problems, breakthrough of the gap between first-order dynamics and higher-order ones demands for more advanced technical developments. In this paper, we explore a Paillier encryption based average consensus algorithm for a network of systems with second-order dynamics, with randomness added to network weights. The conditions for privacy preserving, especially depending on consensus rate, are thoroughly studied with theoretical analysis and numerical verification.

Key words: Network security, secure control, privacy preserving, Paillier encryption, multi-agents, consensus

1 Introduction

In recent years research on network security has attracted great attention and significant progresses have been made in this field. For example, cyber security of power systems, especially on smart grid, is one of the most popular topics; see, e.g., [1–3]. Network security has also been deeply investigated in many other applications including but not limited to cyber security management of industrial systems [4], secure control for resource-limited adversaries [5], relay attacks [6] and cloud computing [7]. There are various techniques for keeping privacy and security of network systems, among which encryption is an effective but challenging method. A typical encryption technique involving trusted third-party can be found in many references such as [8–12]. In particular, there has been a great impetus within systems and control community to propose effective tools to detect and mitigate the effects of cyber threats; see for example [13], [14], [15], [16], [17] and the references listed therein. Moreover, integration of encryption techniques inside control and estimation paradigms, which only deal with encrypted data, becomes an emerging research topic in recent years [18–20].

Consensus law is perhaps one of the most common protocols in networked systems and has been widely studied not only by systems and control theorists, but also by physicists, biologists, sociologists, and mathematicians. In systems and control community, this protocol has been investigated under consensus of a multi-agent system (MAS) topic. The early works on consensus of MAS with first-order dynamics can be found in [21, 22], in both continuous-time and discrete-time settings. Over the past decade, extensive efforts have been devoted to the research on more complicated systems including second-order systems [23, 24], general linear systems [25, 26], and even nonlinear heterogeneous systems [27, 28]. It is known from the history of consensus theory that breakthrough from first-order dynamics to second-order ones was often significant in many scenarios; see a survey paper [29]. It is worth mentioning that the consensus problem of a second-order MAS with time-varying topologies still remains open with many attempts in, e.g., [30, 31], while the same problem for a first-order MAS has been well studied many years ago in [32, 33].

Given its wide range of applications, privacy and security challenges associated with consensus protocol is of great practical and theoretical importance. As consensus protocols require information exchange among agents within a network, if an agent’s state is completely known by others at one time instant, its whole trajectory might be reconstructed using the knowledge of dynamics. In many practical situations, leakage of information is not allowed. Therefore, researchers are interested in preserv-
ing privacy of an agent, typically, keeping its initial states secret from others, during the evolution of network. A common privacy preserving approach is to use obfuscation to mask the true state values by adding random noises to the average consensus process; see, e.g., [34,35]. In this noise-based setting, differential privacy is another important tool for database privacy in computer science [36–38]. The result for privacy preserving maximum consensus can be found in [39] where all agents independently generate and transmit random numbers before sending out their initial states. The noise-based obfuscation techniques prevent forming consensus at the desired average value and achieve consensus in the statistical mean-square sense. To mitigate such effects on network performance, researchers are proposing other secure control protocols. Another common approach is to employ tools from cryptography.

Application of cryptography-based approaches to a decentralized protocol like consensus, especially without a trusted third-party, is challenging. An early result can be found in [40] where privacy is preserved at the cost of depriving participating agents from access to the agreed value. A more complete result has been reported in a recent paper [41] for first-order agents where a homomorphic cryptography-based approach is used to guarantee privacy and security in a decentralized consensus scenario. Also, randomness is added to the coupling weights such that the transmission signals via network are of higher confidentiality. However, in their work, the privacy analysis result is only for first-order systems, where leakage of coupling weights directly leads to disclosure of agents’ states. This phenomenon may not necessarily happen, depending on the network convergence rate, in the case of agents with second-order dynamics.

In this paper, we attempt to propose a secure and privacy preserving average consensus algorithm for second-order multi-agent systems, using a cryptography-based approach as well as random network coupling weights. The research for second-order multi-agent systems has two-fold contributions.

On one hand, a protocol with random network coupling weights unavoidably leads to a network of time-varying weighted topologies. Therefore, the development relies on a consensus algorithm for time-varying topologies. In a discrete-time setting, sufficient and necessary conditions for second-order multi-agent systems were revealed in [24], only for a fixed topology. The result in [42] applies for time-varying topologies, but with a compulsory velocity damping, which results in agent velocities agreeing at zero. The scenarios studied in [43] and [44] cover second-order systems, but only for neutrally stable systems whose eigenvalues are semi-simple with modulus 1. The authors of [45] studied the general scenario as in this paper; however, with very restrictive network connectivity conditions including fully connected, neighbor shared, and at least quasi-neighbor shared networks. A very recent paper for dealing with second-order systems with time-varying topologies can be found in [46], but only in a continuous-time setting. As the cornerstone for establishing a secure and privacy preserving average consensus algorithm, we first establish conditions for achieving consensus among second-order agents with time-varying topology and discrete time dynamics. This is the first contribution which can be regarded as extension of a result in [24].

The second contribution lies in development of an algorithm to preserve agents privacy within the network. The proposed cryptography-based consensus algorithm with random network weights guarantees privacy for an agent connected with more than one neighbors. Moreover, we fully examine the scenarios under which leakage of private information might occur. It is worth mentioning that in a network of second-order agents, the two-dimensional agent states, say position and velocity, are encrypted and transmitted in a lumped quantity. Hence, the secure and privacy preserving analysis becomes totally different from that of first-order systems. Moreover, it turns out that unlike the case of first-order systems, for a second-order system scenario, the possibility of estimating a neighbor’s initial state depends on the network convergence rate, even when the weights are fully known by agents.

The rest of this paper is organized as follows. In Section 2, graph theory and Paillier encryption are briefly introduced. The main results including a consensus algorithm for second-order systems with time-varying topologies and its cryptography based version are proposed in Section 3. The conditions for privacy preserving are also revealed in this section. Numerical examples are presented in Section 4. Finally, some conclusions are drawn in Section 5.

2 Preliminaries

In this section, some preliminary concepts regarding graph theory and Paillier cryptosystem are briefly introduced.

2.1 Graph Theory

Let \( G(\mathbb{V}, \mathbb{E}) \) denote a connected undirected graph. The set of all nodes and edges are presented by \( \mathbb{V} \) and \( \mathbb{E} \), respectively. The elements in \( \mathbb{V} \) are \( \nu_1, \nu_2, \ldots, \nu_N \), where \( N \) is the number of nodes, and the elements of the set \( \mathbb{E} \) are pairs of nodes denoted by \( (\nu_i, \nu_j) \). In a network associated with graph \( G(\mathbb{V}, \mathbb{E}) \), each node \( \nu_i \) represents an agent and each edge \( (\nu_i, \nu_j) \) exists if the agents \( i \) and \( j \) communicate with each other. An adjacency matrix \( \mathcal{A} \) is defined as follows. For every entry \( a_{ij} \), i.e., coupling weight between \( \nu_i \) and \( \nu_j \), if \( (\nu_j, \nu_i) \in \mathbb{E} \), there is \( a_{ij} > 0 \), otherwise \( a_{ij} = 0 \). And the Laplacian matrix is defined as \( L = \{l_{ij}\} \), where \( l_{ij} = -a_{ij}, i \neq j \), and \( l_{ii} = \sum_{j=1, j \neq i}^{N} a_{ij}, \forall \nu_i, \nu_j \in \mathbb{V} \).
2.2 Paillier Encryption

2.2.1 Asymmetric Cryptosystem

In the asymmetric cryptosystem, there are two types of keys, namely, a private key $K^{Prv}$ available only to each individual for decryption and a public key $K^{Pub}$ accessible by all entities only for encryption purposes. Consider two individuals, say Alice and Bob, who intend to communicate in a secure way so that a third party, call it Eve, cannot recover the message. To achieve this, Alice generates a public key $K^{Pub}$ that everyone, perhaps including Eve, knows about. Then before Bob sends a message $m$ to Alice, it encrypts it as $E_A(m)$ using $K^{Pub}$ provided by Alice. However, the only way to decrypt the ciphertext $E_A(m)$ is through the private key $K^{Prv}$, which is only known by Alice. Since Eve does not have access to the private key $P^{Prv}$, information privacy remains intact.

2.2.2 Semi-homomorphic Property of Paillier Cryptosystem

The Paillier encryption exploited in this paper is a certain class of asymmetric cryptosystem that adopts semi-homomorphic encryption techniques [47]. Homomorphic encryption allows operations to be done on ciphertext such that after decryption the results remain equivalent to those obtained from performing some other operations on the associated plaintext. A fully-homomorphic encryption has the property that for an arbitrary function $f$ operating on encrypted data $E(m_i), i = 1, 2, \ldots, n$, which are obtained from encryption of messages $m_i$ under the same public key, there always exists a function $f'$ such that $E(f'(m_1, m_2, \ldots, m_n)) = f(E(m_1), E(m_2), \ldots, E(m_n))$. In a semi-homomorphic cryptosystem, the above property only holds for some limited classes of functions. For instance, Paillier encryption has additive semi-homomorphic property. Under Paillier encryption, the following properties hold,

$$E(m_1) \cdot E(m_2) = E(m_1 + m_2)$$

$$(E(m_1))^n = E(n \cdot m_1),$$

where $m_1$ and $m_2$ are plaintexts and $E$ denotes encryption function.

A brief review of Paillier algorithm is demonstrated in Algorithm 1.

Algorithm 1 Paillier Cryptosystem

**Key Generation:**
1. Generate 2 large prime number $p$, $q$, where $\gcd(pq, (p-1)(q-1)) = 1$.
2. $\lambda = \text{lcm}(p-1, q-1)$, and modulus $n = p \cdot q$.
3. Define $Z_n = \{x | x \in \mathbb{Z}, 0 \leq x < n\}$, $Z_n^* = \{x | x \in \mathbb{Z}, 0 \leq x < n, \gcd(x, n) = 1\}$.
4. Select an integer $\gamma \in Z_{n^2}^*$ that satisfies

$$\gcd(L(g^\gamma \mod n^2), n) = 1,$$

where the function $L$ is defined by

$$L(x) = \frac{x - 1}{n},$$

$$x \in \{x < n^2 | x \equiv 1 \mod n\}.$$

Public key is $(n, g)$. Private key is $(\lambda)$.

**Encryption:**
1. Generate a random integer $r \in (0, n)$.
2. Compute

$$c = g^m \cdot r^n \mod n^2.$$  

**Decryption:**
1. Compute

$$m = \frac{L(c^\lambda \mod n^2)}{L(g^\lambda \mod n^2)} \mod n$$

agent, i.e., its position or velocity, might be leaked to an eavesdropper or a curious neighbor. It is noted that the possibility of leakage due to disclosure of private keys is out of the scope of the current paper and is not considered.

3.1 Second-order Systems

Consider a second-order system with its dynamics represented in discrete-time form as follows

$$p_i^{(k+1)} = p_i^{(k)} + Tv_i^{(k)},$$

$$v_i^{(k+1)} = v_i^{(k)} + Tu_i^{(k)}, \quad i = 1, 2, \ldots, N,$$

where $k$ is the time index, $T$ is the sampling time, and $p_i, v_i \in \mathbb{R}$ are the position and velocity associated with the agent $i$, respectively. Without loss of generality, we let the sampling time be unity throughout this paper. The following control law drives the group of agents in (2) toward reaching consensus asymptotically

$$u_i^{(k)} = \sum_{j \in N_i} \gamma_1 a_{ij}^{(k)} (p_j^{(k)} - p_i^{(k)}) + \gamma_2 \sum_{j \in N_i} a_{ij}^{(k)} (v_j^{(k)} - v_i^{(k)}),$$

where $a_{ij}^{(k)}$'s are entries of adjacency matrix $A$ derived from a connected and undirected network topology, $\gamma_1, \gamma_2.$
\(\gamma_2\) are the coefficients and \(N_i\) is the set of agent \(i\)'s neighbors.

Let
\[
p^{(k)} = \left( p^{(k)}_1, p^{(k)}_2, \ldots, p^{(k)}_N \right)^T,
\]
\[
v^{(k)} = \left( v^{(k)}_1, v^{(k)}_2, \ldots, v^{(k)}_N \right)^T,
\]
\[
x^{(k)} = \left( (p^{(k)})^T, (v^{(k)})^T \right)^T.
\]

With the definition of the Laplacian matrix \(L^{(k)}\), substituting (3) into (2) yields
\[
x^{(k+1)} = F^{(k)}x^{(k)},
\]
where \(F^{(k)} = \left[ \begin{array}{cc} I & \mu I \\ -\gamma_1 L^{(k)} & I_\mu - \gamma_2 L^{(k)} \end{array} \right]\). If the system has a fixed topology, the dynamics in (4) shrinks to
\[
x^{(k+1)} = Fx^{(k)}.
\]

For an undirected and connected topology, the Laplacian matrix \(L\) attains an eigenvalue 0 associated with a right eigenvector \(1_N\) and a left eigenvector \(1_N^\top\); see, e.g., [48]. Thus, one can easily conclude that \(F\) has eigenvalue 1 associated with algebraic multiplicity two, associated with right eigenvectors \((1_N^\top, 0_N^\top)^\top\) and \((0_N^\top, 1_N^\top)^\top\). Moreover, since \(L\) is symmetric, the matrix \(F\) has left eigenvectors \((1_N^\top, 0_N^\top)^\top\) and \((0_N^\top, 1_N^\top)^\top\).

We need the following definitions which are required for the further developments presented in this paper.

**Definition 1** A group of discrete-time second-order agents with dynamics as in (2) is said to reach (asymptotic) consensus if the following property holds for any initial conditions:
\[
\lim_{k \to +\infty} p^{(k)}_i - p^{(k)}_j = 0, \quad \lim_{k \to +\infty} v^{(k)}_i - v^{(k)}_j = 0, \quad \forall i, j \in \mathbb{V}.
\]

The above definition is concerned about consensus as \(k \to \infty\), which is an asymptotic property. To explicitly describe a consensus behavior in finite time, we introduce the following definition.

**Definition 2** A group of discrete-time second-order agents with dynamics as in (2) is said to reach \(\delta\)-practical consensus at time \(k^*_c\) if
\[
|p^{(k)}_j - p^{(k)}_i| < \delta, \quad \forall k > k^*_c \text{ and } i, j \in \mathbb{V}.
\]

It is noted that the above definition demands for global knowledge of positions and/or velocities of agents. Therefore, we introduce the following definition that relies only on the local information.

**Definition 3** In a group of discrete-time second-order agents with dynamics as in (2), two agents, say \(i\) and \(j\), are said to reach local \(\delta\)-agreement at time \(k^*_\alpha\) if
\[
|p^{(k)}_j - p^{(k)}_i| \leq \delta.
\]

We now revisit a lemma from [24] that provides conditions for a group of agents with dynamics as in (2) to reach consensus.

**Lemma 1** Suppose that a group of discrete-time second-order agents with dynamics as in (2) is connected under a fixed topology with the control law (3). Then the closed-loop system in (5) reaches consensus with
\[
\begin{align*}
\gamma_2 > \gamma_1 > 0, \\
\gamma_1 - 2\gamma_2 > \frac{-4}{\mu_i},
\end{align*}
\]

where \(\mu_i\)'s are nonzero eigenvalues of the Laplacian matrix \(L\), if and only if the associated topology is connected.

**Proof:** To be self-contained, the proof in [24] is briefly presented here. Given the connectivity of the topology, the Laplacian matrix of the system has an eigenvalue 0 associated with an eigenvector \(1_N\). The condition (8) ensures that the rest eigenvalues of \(L\) are negative, which makes the remaining eigenvalues of \(F\) stay within the unit circle. Let \(J\) be the Jordan canonical form of \(F\) associated with an invertible matrix \(P\); then we have
\[
F = JPJ^{-1} = P\begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0_{2 \times (2N - 2)} \\ 0_{(2N - 2) \times 2} \end{bmatrix} P^{-1},
\]
where \(J\) is a Jordan canonical form matrix containing all the eigenvalues of \(J\) except an eigenvalue 1 with algebraic multiplicity two. The vectors \(\frac{1}{\sqrt{N}}(1_N^\top, 0_N^\top)^\top\) and \(\frac{1}{\sqrt{N}}(0_N^\top, 1_N^\top)^\top\) are the first and second rows of \(P\), respectively. One should notice that all eigenvalues of \(J\) are smaller than one and \(\lim_{k \to \infty} \bar{J}^k = 0\). Then by substituting (9) into (5), we calculate the norm
\[
\lim_{k \to \infty} \left\| \begin{bmatrix} p^{(k)} \\ v^{(k)} \end{bmatrix} - \begin{bmatrix} \frac{1}{N} \sum_{j=1}^N (p^{(0)}_j + k v^{(0)}_j) \\ \frac{1}{N} \sum_{j=1}^N v^{(0)}_j \end{bmatrix} \otimes 1_N \right\| = 0.
\]

**Theorem 1** There exists an admissible variation range \(\delta_{A} > 0\) such that all agents in the aggregated model (4)
under an undirected graph topology reach consensus if the following conditions are satisfied

\begin{align}
\begin{cases}
\|A^{(k)} - A^{(0)}\| < \delta_A, \forall k \geq 0, \\
\gamma_2 > \gamma_1 > 0, \\
\gamma_1 - 2\gamma_2 > -\frac{4}{\mu_{(i)}},
\end{cases}
\end{align}

(11)

where \(\mu_{(i)}\) are nonzero eigenvalues of the Laplacian matrix \(L^{(0)}\), \(A\) is the adjacency of topology matrix, and \(\|A\|\) denotes the max norm of \(A\).

**Proof:** Firstly we calculate the Jordan canonical form of the matrix \(F^{(0)}\):

\[
F^{(0)} = PJP^{-1}
\]

\[
= P \begin{bmatrix}
1 & 1 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
0_{2 \times (2N-2)}
\end{bmatrix} P^{-1}.
\]

(12)

Given that the matrices \(F^{(k)}\) and \(F^{(0)}\) share common eigenvectors, by using the same matrix \(P\), we can decompose \(F^{(k)}\) as

\[
F^{(k)} = PH^{(k)}P^{-1}
\]

\[
= P \begin{bmatrix}
1 & 1 \\
0 & 1
\end{bmatrix} \begin{bmatrix}
0_{2 \times (2N-2)}
\end{bmatrix} P^{-1},
\]

(13)

where \(H^{(0)} = J\) and \(\tilde{H}^{(0)} = \tilde{J}\). One should note that using the same \(P\), for \(k = 0, 1, \ldots\), makes \(H^{(k)}\)'s not necessarily be Jordan blocks when \(k \geq 1\). However, given the structure of the matrices \(F^{(k)}\)'s, all of them share two common left eigenvectors and two right eigenvectors, therefore \(H^{(k)}\)'s have a common block

\[
\begin{bmatrix}
1 & 1 \\
0 & 1
\end{bmatrix}.
\]

We now apply the following coordinate transformation

\[
\bar{x}^{(k)} = P^{-1}x^{(k)}
\]

(14)

and let \(\bar{x}^{(k)} = \begin{bmatrix} \sigma_k \\ \zeta_k \end{bmatrix}\) for \(\sigma_k \in \mathbb{R}^2\) and \(\zeta_k \in \mathbb{R}^{2N-2}\).

One should note that consensus is reached if \(\zeta_k \to 0\) as \(k \to \infty\).

Consider a Lyapunov function

\[
Y_k = \zeta_k^TQ\zeta_k.
\]

(15)

With the help of (4), (13), (14) and (15), one can write

\[
Y_{k+1} - Y_k = \zeta_{k+1}^TQ\zeta_{k+1} - \zeta_k^TQ\zeta_k
\]

\[
= \zeta_k^T((H^{(k)}R^{(k)}Q\tilde{H}^{(k)} - Q)\zeta_k.
\]

(16)

According to Lemma 1, the conditions (11) ensure that all eigenvalues of \(H^{(0)}\) are inside the unit circle, which means there exists a positive definite matrix \(Q\) such that

\[
R^{(0)} = ((J)^TQ\tilde{J} - Q)
\]

is negative definite. As the change in eigenvalues of a matrix is a continuous function of its entries, one can conclude that \(R^{(k)}\) is also a negative definite matrix for a slight variation in entries of the matrix \(A\).

### 3.2 Privacy-preserving Consensus In Undirected Networks

First of all, preserving an agent’s privacy throughout the paper is defined as maintaining its initial states hidden from other agents. In this subsection, we introduce a strategy for preserving privacy among a group of agents with dynamics as in (2) under an undirected network topology. In particular, we provide a paradigm that manages encryption and information exchange policies to maintain privacy of agents’ information from their neighbors while achieving consensus among them.

#### 3.2.1 Confidential Strategy

To explain our proposed method in this subsection, we focus on an agent, say Alice, which communicates with other agents within its neighbor set under an undirected topology. One should note that all agents are employing the same protocol as in (3), if there is no state information leakage happening in process of calculating \(u_A^{(k)}\), i.e., Alice’s control input, the privacy of all agents remains intact.

The information exchange strategy is the same between Alice and its neighbors, thus without loss of generality, we only need to examine the interaction between Alice and one of its neighbors, say Bob.

In the following, we examine the possibility of information leakage from Bob to Alice in the process of computation of \(u_A^{(k)}\). To this end, let us expand Alice’s command input as follows

\[
u_A^{(k)} = \gamma_1a_{AB}^{(k)}(p_B^{(k)} - p_A^{(k)}) + \gamma_2a_{AB}^{(k)}(v_B^{(k)} - v_A^{(k)}) + \sum_{j \in N_A \setminus \{Bob\}} \gamma_1a_{AJ}^{(k)}(p_j^{(k)} - p_A^{(k)}) + \gamma_2a_{AJ}^{(k)}(v_j^{(k)} - v_A^{(k)}),
\]

where \(N_A\) denotes set of Alice’s neighbors and \(a_{AB}^{(k)}\) is an entry of the adjacency matrix \(A\) corresponding to Alice and Bob. Then Bob’s contribution in \(u_A^{(k)}\) is

\[
u_{AB}^{(k)} = \gamma_1a_{AB}^{(k)}(p_B^{(k)} - p_A^{(k)}) + \gamma_2a_{AB}^{(k)}(v_B^{(k)} - v_A^{(k)}).
\]

(17)

Similarly, the contribution of Alice to \(u_B^{(k)}\) can be written as \(u_{BA}^{(k)}\).
 Remark 1 According to (3) and (17), $u_{AB}^{(k)}$ is necessary information that Alice requires from Bob to compute $u_A^{(k)}$, which is the input to update $v_A^{(k)}$. In this scenario, if the states of Bob can be deduced from $u_{AB}$, its privacy cannot be guaranteed under any secure communication protocol.

For a first-order system, $v_{AB}^{(k)}$ in (17) reduces to

$$u_{AB}^{(k)} = a_{AB}^{(k)}(p_{B}^{(k)} - p_{A}^{(k)})$$

and the state of Bob can be directly computed from $u_{AB}^{(k)}$ with $a_{AB}^{(k)}$ known, i.e.,

$$p_{B}^{(k)} = u_{AB}^{(k)} + p_{A}^{(k)}.$$

The situation for a second-order system is more complicated as $u_{AB}^{(k)}$ in (17) contains lumped quantity of position and velocity. Nevertheless, the computation of Bob’s states is still possible when it has a sole neighbor. More specifically, at the two steps of $k = 0$ and $k = 1$, the two messages received from Bob by Alice are as follows:

$$u_{AB}^{(0)} = \gamma_1 a_{AB}^{(0)}(p_{B}^{(0)} - p_{A}^{(0)}) + \gamma_2 a_{AB}^{(0)}(v_{B}^{(0)} - v_{A}^{(0)}),$$

$$u_{AB}^{(1)} = \gamma_1 a_{AB}^{(1)}(p_{B}^{(1)} - p_{A}^{(1)}) + \gamma_2 a_{AB}^{(1)}(v_{B}^{(1)} - v_{A}^{(1)}).$$

Also, Bob’s states obey the second-order dynamics

$$p_{B}^{(1)} = p_{B}^{(0)} + v_{B}^{(0)},$$

$$v_{B}^{(1)} = v_{B}^{(0)} + u_{B}^{(0)}.$$

As the topology is undirected and Bob has one sole neighbor, Alice is able to measure

$$v_{B}^{(0)} = u_{BA}^{(0)} = -u_{AB}^{(0)}.$$

The other four variables, i.e., $p_{B}^{(0)}, p_{B}^{(1)}, v_{B}^{(0)}$ and $v_{B}^{(1)}$, can be computed from the set of four equations in (18) and (19). In particular, Bob’s initial states are

$$p_{B}^{(0)} = p_{A}^{(0)} - \frac{\gamma_1 + \gamma_2 a_{AB}^{(0)}(p_{B}^{(0)} - p_{A}^{(0)})}{\gamma_1^2 a_{AB}^{(0)} + \frac{\gamma_2^2}{\gamma_1^2} a_{AB}^{(1)}},$$

$$v_{B}^{(0)} = v_{A}^{(0)} - \frac{1}{\gamma_1} \left( \frac{u_{AB}^{(0)}}{a_{AB}^{(0)}} - \frac{u_{AB}^{(k)}}{a_{AB}^{(k)}} \right) + \frac{\gamma_2}{\gamma_1} (v_{A}^{(1)} - v_{A}^{(0)}).$$

Thus, Bob’s privacy cannot be maintained irrelevant of which kind of security protocol is applied.

If Bob has more than one neighbors, the above computation becomes invalid. For this scenario, we establish

privacy preserving conditions in the following theorem, for a group of agents with second-order dynamics under undirected topologies. Also, it is noted that the weights between agents and its neighbors are public information. In order to analyze the privacy of agents’ states during consensus process, let $\hat{p}_{B}^{(k)}$ and $\hat{v}_{B}^{(k)}$ be the estimation values of Bob’s position and velocity, calculated by Alice. Then the estimation errors of Bob’s initial states are defined as follows.

$$\varepsilon_p = \hat{p}_{B}^{(0)} - p_{B}^{(0)},$$

$$\varepsilon_v = \hat{v}_{B}^{(0)} - v_{B}^{(0)}.$$

Theorem 2 Consider two agents with dynamics as in (2), called Alice and Bob, which are connected via an undirected network topology under the control law (3). Suppose $a_{AB}^{(k)}$ is public information. Then, when the local $\delta$-agreement between Alice and Bob is reached at time $k_a$, by collecting $u_{AB}^{(k)}$, $p_{A}^{(k)}$ and $v_{A}^{(k)}$, $k = 1, 2, \cdots, k_a$, Bob’s initial position and velocity can be estimated by Alice with errors $\varepsilon_p$ and $\varepsilon_v$, respectively, which satisfy

$$|\varepsilon_p| \leq \left( \frac{\gamma_2}{\gamma_2 - \gamma_1} \right) k_a \delta,$$

$$|\varepsilon_v| \leq \frac{\gamma_2^k}{(\gamma_2 - \gamma_1)^k} k_a \delta.$$

Proof: According to the control law (3), Alice obtains a set of equations based on the information $u_{AB}^{(k)}$ collected from $k = 0$ to $k = k_a - 1$, i.e.,

$$u_{AB}^{(0)} = \gamma_1 a_{AB}^{(0)}(p_{B}^{(0)} - p_{A}^{(0)}) + \gamma_2 a_{AB}^{(0)}(v_{B}^{(0)} - v_{A}^{(0)}),$$

$$\vdots$$

$$u_{AB}^{(k_a-1)} = \gamma_1 a_{AB}^{(k_a-1)}(p_{B}^{(k_a-1)} - p_{A}^{(k_a-1)}) + \gamma_2 a_{AB}^{(k_a-1)}(v_{B}^{(k_a-1)} - v_{A}^{(k_a-1)}).$$

Next, with the knowledge of (2), Alice can also construct another set of equations as below

$$p_{B}^{(1)} = p_{B}^{(0)} + v_{B}^{(0)},$$

$$\vdots$$

$$p_{B}^{(k_a)} = p_{B}^{(k_a-1)} + v_{B}^{(k_a-1)}.$$

At time $k_a$, define

$$\delta_p = p_{B}^{(k_a)} - \hat{p}_{B}^{(k_a)}.$$

By setting $\hat{p}_{B}^{(k_a)} = p_{B}^{(k_a)}$, one has $|\delta_p| \leq \delta$ by Definition 3.
Substituting (26) into the last equation of (25) provides

\[
\begin{align*}
\dot{p}_B^{(k)} &= \dot{p}_B^{(0)} + v_B^{(0)}, \\
\vdots \\
\dot{p}_B^{(k_a-1)} &= \dot{p}_B^{(k_a-2)} + v_B^{(k_a-2)}, \\
\dot{p}_B^{(k_a)} &= \dot{p}_B^{(k_a-1)} + v_B^{(k_a-1)} - \delta_p.
\end{align*}
\]

(27)

Next, by eliminating \(v_B^{(k)}\), \(k = 0, \ldots, k_a - 1\), in (24) using (25), one obtains the following set of equations, for \(k = 0, \ldots, k_a - 1\),

\[
(\gamma_2 - \gamma_1)p_B^{(k)} = \gamma_2p_B^{(k+1)} - \gamma_1p_A^{(k)} - \gamma_2v_A^{(k)} - \frac{u_{AB}}{a_{AB}}
\]

(28)

Given the above equation by invoking (26), we can express the position and velocity of Bob as

\[
p_B^{(k)} = \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a - k}p_B^{(k_a)} + \sum_{T=0}^{k_a - 1} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^T \varphi_{k+T} + \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a - k} \delta_p,
\]

\[
v_B^{(k)} = \frac{\gamma_1}{\gamma_2 - \gamma_1} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a - k}p_B^{(k_a)} - \gamma_2v_A^{(k)} - \frac{u_{AB}}{a_{AB}}
\]

(29)

where

\[
\varphi_k = \frac{1}{\gamma_1 - \gamma_2} \left[ \gamma_1p_A^{(k)} + \gamma_2v_A^{(k)} + \frac{u_{AB}}{a_{AB}} \right].
\]

By letting \(k = 0\) in (29), Bob’s initial states, i.e., its position and velocity, can be obtained as

\[
p_B^{(0)} = \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a}p_B^{(k_a)} + \sum_{k=0}^{k_a - 1} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^k \varphi_k + \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a} \delta_p,
\]

\[
v_B^{(0)} = -\frac{\gamma_1}{\gamma_2} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a}p_B^{(k_a)} - \gamma_2v_A^{(0)} - \frac{u_{AB}}{a_{AB}}
\]

(30)

Alice can estimates Bob’s initial states, i.e., \(p_B^{(0)}\) and \(v_B^{(0)}\) through the following estimation law :

\[
\dot{\hat{p}}_B^{(0)} = \frac{\gamma_2}{\gamma_2 - \gamma_1} \hat{p}_B^{(k_a)} + \sum_{k=0}^{k_a - 1} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^k \varphi_k,
\]

\[
\dot{\hat{v}}_B^{(0)} = -\frac{\gamma_1}{\gamma_2} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a}p_B^{(k_a)} - \gamma_2v_A^{(0)} - \frac{u_{AB}}{a_{AB}}
\]

(31)

Finally, with respect to (22), (30) and (31), the estimation error of Bob’s states computed by Alice are as follows,

\[
\varepsilon_p = \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a} \delta_p,
\]

\[
\varepsilon_v = \frac{\gamma_1}{\gamma_2 - \gamma_1} \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^{k_a} \delta_p.
\]

(32)

Since \(|\delta_p| < \delta\), (32) implies (23) and the proof is finished.

**Remark 2** For the sequence \(k = 0, 1, \ldots, \) one can define

\[
\delta(k) = |p_B^{(k)} - p_B^{(k)}|.
\]

Obviously, Alice and Bob reaches local \(\delta(k)\)-agreement at time \(k\). Theorem 2 claims that the estimation error of Bob’s initial position at time \(k\) is bounded by

\[
|\varepsilon_p(k)| \leq \left(\frac{\gamma_2}{\gamma_2 - \gamma_1}\right)^k \delta(k) = \frac{\delta(k)}{(\gamma_2 - \gamma_1)^k}.
\]

In a consensus process, one has \(\delta(k) \to 0\) as \(k \to \infty\). If the consensus convergence is sufficiently fast in the sense of

\[
\frac{\delta(k)}{(\gamma_2 - \gamma_1)^k} \to 0, \quad \text{as} \quad k \to \infty,
\]

Alice may estimate Bob’s initial position with a sufficiently small estimation error. However, if the consensus convergence is not fast, sufficiently precise estimation of Bob’s initial position becomes impossible, that is, Bob’s privacy is preserved. The same arguments also hold for estimation of Bob’s initial velocity.

**Remark 3** It is worthwhile noting that the computation of estimation error in (23) depends on achieving local \(\delta\)-agreement at time \(k_a\). However, in a privacy-preserving consensus states of neighbors are secret information, which means \(k_a\) cannot be obtained directly by agents. Consider the two agents with specifications stated in Theorem 2. Alice has only access to its own states and Bob’s message \(u_{AB}\). Since the states of Bob remain hidden from Alice, it becomes impossible for it to directly attain information of \(k_a\) even with the knowledge of permissible error \(\delta\). Furthermore, in a second-order system, \(u_{AB}^{(k)} = 0\) does not imply \(|p_B^{(k)} - p_A^{(k)}| = 0\)
or $|v_B^{(k)} - v_A^{(k)}| = 0$. However, it is possible to indirectly estimate the value of $k_a$ using the $u_{AB}^{(k)}$.

3.2.2 Exchanging Information And Operation

Theorem 2 demonstrates that in an undirected topology the knowledge of link weights, i.e., $a_{ij}^{(k)}$, relating two neighbors, say Alice and Bob, enable Alice to reconstruct the states of Bob by exploiting the collected inputs $v_{AB}^{(k)}$ from Bob and its own states, provided that the consensus convergence rate is sufficiently large. To overcome this shortcoming, we exploit a methodology initially introduced by [41] for networks of agents with first-order dynamics as in Theorem 3 of notation we omit the parameter $k$ to system to maintain the privacy of information between Alice and Bob. Moreover, according to (3), in an undirected network it holds that $a_{AB}^{(k)} = a_{BA}^{(k)}$. Then Alice can calculate Bob’s state $v_B^{(k)}$ using

$$v_B^{(k)} = v_A^{(k)} + \sum_{i=k}^{k-1} u_{AB}^{(i)}$$

This holds irrespective of the encryption method Bob and Alice exploiting in their communication. Now if Bob has more than one neighbors, since the decoupled term $u_{AB}^{(k)}$ is only kept by Bob, substituting $a_{AB}^{(k)}$ into equation

Algorithm 2 Information exchange in undirected networks

**Preparation (Alice):**

1. At initial time $k = 0$, generate a pair of public key $K_A^{pub}$ and private key $K_A^{priv}$, then send $K_A^{pub}$ to all its neighbors, including Bob.
2. At time $k$, generate a random number $a_A \in (\sqrt{a_{AB}^{(0)} - \delta_A}, \sqrt{a_{AB}^{(0)} + \delta_A})$.

**Preparation (Bob):**

1. At time $k$, generate a random number $a_B \in (\sqrt{a_{AB}^{(0)} - \delta_A}, \sqrt{a_{AB}^{(0)} + \delta_A})$.

**Step 1 (Alice):**

1.1) Encrypt position: $p_A \rightarrow -p_A \xrightarrow{a_A} E_A(-p_A) \text{ sent to } Bob$.

1.2) Encrypt velocity: $v_A \rightarrow -v_A \xrightarrow{a_A} E_A(-v_A) \text{ sent to } Bob$.

**Step 2 (Bob):**

2.1) Operate position: $p_B \rightarrow -p_B \xrightarrow{a_B} E_A(p_B) \rightarrow E_A(p_B + p_A)$.

2.2) Operate velocity: $v_B \rightarrow -v_B \xrightarrow{a_B} E_A(v_B) \rightarrow E_A(v_B - v_A)$.

2.3) Combine $p$ and $v$: $E_A(\gamma a_B \cdot (p_B - p_A) + \gamma_2 a_B \cdot (v_B - v_A)) = E_A(\gamma a_B \cdot (p_B - p_A) + \gamma_2 a_B \cdot (v_B - v_A)) \xrightarrow{a_B} \gamma_1 a_{AB} \cdot (p_B - p_A) + \gamma_2 a_B \cdot (v_B - v_A)$.

**Step 3 (Alice):**

Decrypt and operate:

$E_A(\gamma a_B \cdot (p_B - p_A) + \gamma_2 a_B \cdot (v_B - v_A)) \xrightarrow{a_B} \gamma_1 a_{AB} \cdot (p_B - p_A) + \gamma_2 a_B \cdot (v_B - v_A) = u_{AB}$.

(24) with $a_A^{(k)}, a_B^{(k)}$ provides Alice more unknowns but no additional equations. This makes the set of equations collected by Alice be unsolvable. Therefore privacy of Bob’s states remains intact. This completes the proof.

Theorems 2 and 3 provide the conditions for privacy preserving consensus of second-order systems, in the cases of known coupling weights and unknown coupling weights, respectively. The corresponding conditions for first-order systems have been studied in [41]. Table 1 outlines different scenarios that might raise depending on the agents’ dynamics, their number of neighbors and availability of coupling weights information. It demonstrates the possible violation of individual’s privacy and time required for that to happen.

4 Simulation

In this section, some numerical examples are presented to demonstrate how the privacy is preserved under Algorithm 2. We implement the Paillier method written in C language [49]. During the simulation, modular bits used for generating keys of Paillier encryption are set to 64.

We consider the topology in Fig. 1 with the associated
Depending on consensus convergence speed, the consensus is not reached but since local $\delta$-agreement is achieved between agents A and B for a small $\delta$, the estimated values of agent B’s initial states are accurate.

As expected by Theorem 2, in Fig. 3, agent A estimates B’s initial states with error values which converge to 0 as time $k \to \infty$. It occurs in a consensus process with a sufficiently fast convergence rate. Next, with all the coupling weights $a_{ij}B$ multiplied by a factor 0.8, the consensus convergence rate reduces accordingly. In this case, estimation of B’s initial states is inaccurate as illustrated in Fig. 4. That is, B’s privacy still remains intact.

Finally, as far as the computation complexity of the proposed scheme is concerned, it was recorded that the average computation time for each step of simulation is 174.9 ms on a laptop with 2.9 GHz Intel Core i7 Dell laptop using Matlab R2017a, which runs Linux on a Virtual Machine. Each step contains 12 encryption processes and 8 decryption processes associated with all 4 agents.

5 Conclusion

In this paper, we have provided a cryptography-based secure consensus protocol for networks of second-order agents. The proposed protocol achieved consensus asymptotically among all agents while keeping the privacy of individuals intact. Furthermore, the communication weights information is available to all agents, we have performed a full privacy analysis and shown that violation of individual’s privacy depends on the network convergence rate.

References

[1] W. Wang and Z. Lu, “Cyber security in the smart grid: Survey and challenges,” Computer Networks, vol. 57, no. 5, pp. 1344–1371, 2013.
[2] Y. Yan, Y. Qian, H. Sharif, and D. Tipper, “A survey on cyber security for smart grid communications,” IEEE Communications Surveys and Tutorials, vol. 14, no. 4, pp. 998–1010, 2012.
[3] J. Liu, Y. Xiao, S. Li, W. Liang, and C. P. Chen, “Cyber security and privacy issues in smart grids,” IEEE Communications Surveys & Tutorials, vol. 14, no. 4, pp. 981–997, 2012.
[4] W. Knowles, D. Prince, D. Hutchison, J. F. P. Disso, and K. Jones, “A survey of cyber security management in industrial control systems,” International Journal of Critical Infrastructure Protection, vol. 9, pp. 52–80, 2015.
[5] A. Teixeira, I. Shames, H. Sandberg, and K. H. Johansson, “A secure control framework for resource-limited adversaries,” Automatica, vol. 51, pp. 135–148, 2015.
[6] Y. Mo and B. Sinopoli, “Secure control against replay attacks,” in Communication, Control, and Computing, 2009. Allerton 2009. 47th Annual Allerton Conference on. IEEE, 2009, pp. 911–918.
[7] R. L. Krutz and R. D. Vines, Cloud security: A comprehensive guide to secure cloud computing. Wiley Publishing, 2010.
[42] P. Lin and Y. Jia, “Consensus of second-order discrete-time multi-agent systems with nonuniform time-delays and dynamically changing topologies,” *Automatica*, vol. 45, no. 9, pp. 2154 – 2158, 2009.

[43] Y. Su and J. Huang, “Two consensus problems for discrete-time multi-agent systems with switching network topology,” *Automatica*, vol. 48, no. 9, pp. 1988 – 1997, 2012.

[44] J. Huang, “The consensus for discrete-time linear multi-agent systems under directed switching networks,” *IEEE Transactions on Automatic Control*, vol. 62, no. 8, pp. 4086 – 4092, 2017.

[45] Y. Chen, J. L, X. Yu, and Z. Lin, “Consensus of discrete-time second-order multiagent systems based on infinite products of general stochastic matrices,” *SIAM Journal on Control and Optimization*, vol. 51, no. 4, pp. 3274–3301, 2013.

[46] H. Meng, Z. Chen, and R. Middleton, “Consensus of multi-agents in switching networks using input-to-state stability of switched systems,” *IEEE Transactions on Automatic Control*, 2018.

[47] P. Paillier, “Public-key cryptosystems based on composite degree residuosity classes,” in *Advances in Cryptology — EUROCRYPT ’99*, J. Stern, Ed. Berlin, Heidelberg: Springer Berlin Heidelberg, 1999, pp. 223–238.

[48] C. Godsil and G. F. Royle, *Algebraic Graph Theory*. Springer Science & Business Media, 2013, vol. 207.

[49] J. Bethencourt, “Paillier library,” 2006. [Online]. Available: http://hms.isi.jhu.edu/acsc/libpaillier/

Fig. 2. Exchanged information under Algorithm 2.

(a) States of agent A.

(b) Encrypted states of agent A as in step 1.1 and 1.2 of Algorithm 2.

(c) Message sent from agent B to A as in Step 2.3 of Algorithm 2.

(d) Decrypted version of the message sent from B to A as in step 3 of Algorithm 2.
Fig. 3. All four agents’ position and velocity trajectories and estimation of agent B’s initial states calculated by agent A in a fast consensus network.

Fig. 4. All four agents’ position and velocity trajectories and estimation of agent B’s initial states calculated by agent A in a slow consensus network.