Taylor & Francis
Taylor & Francis Group

Check for updates

On privacy vs. cooperation in multi-agent systems

Vaibhav Katewa[a], Fabio Pasqualetti[b] and Vijay Gupta[a]

[a]Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, USA; [b]Department of Mechanical Engineering, University of California at Riverside, Riverside, CA, USA

ABSTRACT

This paper considers distributed systems arising when multiple agents cooperatively solve a quadratic optimisation problem. To maintain privacy of their states over time, agents implement a noise-adding mechanism according to the classic differential privacy framework. We characterise how the noise due to the privacy mechanism degrades the performance of the multi-agent system. Interestingly, we show that depending on the desired level of privacy (and thus noise), the system performance is optimised by reducing the level of cooperation among the agents. The notion of cooperation level, which is formally introduced and defined in the paper, models the trust of an agent towards the information received from neighbouring agents. For the prototypical examples of consensus and centroidal Voronoi tessellations, we are able to characterise the optimum cooperation level that maximises the system performance while ensuring a desired privacy level. Our results suggest that for the class of problems we study, and in fact for a broad class of multi-agent systems, it is always beneficial for the agents to reduce their cooperation level when the privacy level increases.

1. Introduction

Several problems in control theory, optimisation and robotics require cooperation among multiple agents. Prototypical examples include consensus (Du, Wen, Cheng, He, & Jia, 2016; Olfati-Saber, Fax, & Murray, 2007), flocking (Olfati-Saber, 2006), formation control (Raffard, Tomlin, & Boyd, 2004), coverage control (Cortes, Martinez, Karatas, & Bullo, 2004) and distributed optimisation (Nedic & Ozdaglar, 2009; Terelius, Topcu, & Murray, 2011). Typically, cooperation requires information exchange, which may lead to leakage of private information with undesirable consequences. For example, in smart metering systems where users send their energy consumption data for power network optimisation, the data can reveal information about their personal lives, such as daily schedules, etc. (McDaniel & McLaughlin, 2009; United States, 2010). In autonomous vehicle scenarios where the vehicles communicate and share their position/velocity data, it can reveal their past or future travel plans. Even if the agents are trustworthy, a possibility exists for an intruder to eavesdrop on the messages exchanged among the agents and gather their private information. Privacy concerns have recently been addressed by introducing dedicated mechanisms as part of the cooperation protocol (Hale & Egerstedt, 2015; Han, Topcu, & Pappas, 2014, 2017; Hsu, Roth, Roughgarden, & Ullman, 2014; Huang, Mitra, & Dullerud, 2012; Huang, Mitra, & Vaidya, 2015; Le Ny & Pappas, 2014). In most of these privacy mechanisms, each agent deliberately adds noise to the data communicated to other agents, thereby preventing them (or an eavesdropper) from recovering the sensitive data of individual agents by accurately processing the distorted messages.

Most of the recently proposed privacy mechanisms are based on techniques originally developed to protect static databases and usually degrade the performance when applied to dynamical systems. For instance, in applications involving wide area control of power grids, adding noise to the data may result in loss of stability. In distributed systems, if agents use noisy information from other agents to update their own state, the resulting behaviour differs from the desired one. Furthermore, both due to the dynamical nature of the system and the fact that information originating from one agent may traverse to another agent through multiple paths in distributed systems, the noise introduced by the agents at one time step can adversely affect the state evolution of a multi-agent system multiple times in the future. Note that the second reason arises because of the cooperative nature of the system where one agent uses information from other agents. Thus, intuitively, there should be a trade-off between the 'cooperation level' and performance in a distributed system when the agents are trying to keep their information private. If the noise level

introduced to maintain privacy is too high, then cooperation might even impede the system functionality. On the other hand, if the agents do not transmit any information to each other, perfect privacy is achieved, at the expense of the benefits of cooperation. In this paper, we address an outstanding and important question whether cooperation leads to improved performance in the presence of a privacy mechanism, and whether a fundamental trade-off exists between the two.

We consider a scenario where the objective of the agents is to cooperatively minimise a common quadratic cost function of their states by sharing their state information among each other. Several problems such as consensus and formation control fall into this class. In addition, the agents wish to keep their states private during this process. We propose a noise-adding privacy mechanism for the agents to keep their states private over time. Note that we focus on *privacy of the agents' state trajectory*, rather than the state at any specific time (as in Hale & Egerstedt, 2015). We adopt the differential privacy (DP) framework originally proposed by Dwork (2006), and later extended to dynamical systems in Le Ny and Pappas (2014). We characterise the noise level ensuring a desired level of privacy.

Next, we introduce a method for the agents to adapt their cooperation level in response to the privacy noise. In many scenarios, in addition to optimising a global cost, the agents also have individual goals that do not require cooperation. For example, in intelligent transport systems, vehicles cooperate to reduce road congestion while each of them also wants to reduce its own travel time (congestion game in Rosenthal (1973)). Individual objectives also exist in multi-objective optimisation problems, wherein multiple conflicting goals are considered and in optimisation problems with separable cost functions. When the agents wish to remain private (by sharing noisy data), it is intuitive that they should cooperate less and focus more on their individual goals. Thus, the cooperation level can be characterised based on whether the agents are willing to cooperatively minimise the global cost, or they want to selfishly minimise their individual costs. We formalise this notion by defining a new cost that is a convex combination of the global and individual costs, wherein the weighing factor represents the *cooperation level*. We then characterise the combined effect of cooperation level and privacy noise on the system performance.

Related work: Several secure multi-party computation schemes exist in the literature which compute a function of agents' variables while keeping them private (Lindell & Pinkas, 2009; Orlandi, 2011). However, in these schemes, there always exists a possibility that some agent(s) obtain auxiliary information and use it to infer other agents'

private variables. Moreover, majority of agents can collude to infer the remaining agents' sensitive information. To address these issues, we use the DP framework in this work. DP abstracts away from any auxiliary information that the agents might have and it is also resilient to post processing of data (Dwork, 2011; Le Ny & Pappas, 2014).

Many recent studies have proposed privacy mechanisms for multi-agent systems. In Huang et al. (2012), the authors present a differentially private consensus algorithm that protects the initial state of the agents, and illustrate privacy vs. accuracy trade-off. In Mo and Murray (2017), a private consensus algorithm is developed using correlated noises, which achieves perfect accuracy. We also study the consensus algorithm as an example and our DP mechanism is similar to Huang et al. (2012). However, we have a different goal of analysing the cooperation vs privacy trade-off. Some papers have addressed privacy issues in optimisation problems. In Huang et al. (2015) and Han et al. (2014), the authors study distributed convex optimisation and optimisation with piecewise affine objectives, respectively, and develop DP mechanisms to keep the agents' cost functions private. In Han et al. (2017) and Hsu et al. (2014) , the authors develop DP mechanisms to keep constraints of the agents private in optimisation problems with convex and linear objectives, respectively. In contrast, our goal is to keep private the entire state trajectories of the agents in a quadratic optimisation problem. In Hale and Egerstedt (2015), the authors develop DP mechanisms to keep the agents' state trajectories private in a convex optimisation problem with nonlinear constraints.

All of these works develop privacy mechanisms and analyse their effect on the system performance in terms of sub-optimality, accuracy, convergence, etc. Thus, they are primarily concerned with the privacy vs. performance trade-off. In contrast, we study a privacy vs. cooperation trade-off by simultaneously characterising the effect of cooperation and privacy noise on the system performance. We develop a privacy mechanism similar to these papers, but we address fundamentally different questions such as: (1) How does the system performance change if the agents vary the amount of cooperation among each other? (2) For a higher privacy level, is it beneficial for the agents to reduce cooperation? We answer these questions by developing a framework through which the agents can vary their cooperation level. To the best of our knowledge, our analysis is novel, and it highlights an important but previously unidentified trade-off in multi-agent systems.

The cooperation level in our framework can be viewed as a weighting factor for the noisy state information received from the neighbours and used to update the agents' states. Related works include Rajagopal and

Wainwright (2011), in which the authors analyse consensus in the presence of noise, and show that almost sure convergence can be guaranteed by using time decaying weighing factor in the updates. In Xiao, Boyd, and Kim (2007), the authors find the optimal edge weights for the consensus problem that minimise the expected deviation among the agents. These works are specifically developed for the consensus algorithm, and may not work for other problems. In contrast, to elucidate the relation between the cooperation and privacy levels in multi-agent systems, we develop techniques that are applicable to more general quadratic optimisation problems and not only limited to consensus.

Contributions: The contribution of this paper is threefold. First, we consider a general class of multi-agent systems arising from the solution of quadratic optimisation problems via distributed computation. We propose a noise-adding privacy mechanism for the agents to solve the optimisation problem while maintaining privacy of their states over time. We analytically quantify the effect of the privacy noise on the system performance. Second, we present a novel method to introduce the notion of 'cooperation level' in cooperative multi-agent systems, as a weighting factor by which the agents weigh the system cost vs. their individual costs. We show that, due to the privacy noise, the performance of the distributed system may improve when reducing the cooperation level among the agents. In fact, we simultaneously characterise the effect of cooperation and privacy levels on the system performance, and show that a fundamental trade-off exists between the two in multi-agent systems. Third and finally, we illustrate our results through the problems of consensus and one-dimensional (1D) Voronoi tessellation. In both cases, we analytically characterise the effect of privacy and cooperation levels on the system cost, and show (by simulation) that an optimal cooperation level exists to maximise the system performance for a desired privacy level. Our results mathematically support the intuition that the optimal cooperation level should decrease if the privacy level increases.

Paper organisation: The rest of the paper is organised as follows. Section 2 presents the quadratic optimisation set-up, the privacy mechanism and its effect on the system performance. Section 3 presents a framework to include a cooperation parameter in the problem and quantifies the performance as a function of privacy noise and agents' cooperation level. Section 4 illustrates our findings by studying the consensus and centroidal Voronoi tessellation (CVT) problems. Finally, Section 5 concludes the paper.

Mathematical notation: The following notation will be adopted throughout the paper. $\|.\|$ denotes the euclidean norm of a vector, or the induced 2-norm of a matrix.

$y[0:\infty]$ denotes an infinite sequence/trajectory. The truncated version of y up to time $T \in \mathbb{N}$ is denoted by $y[0:T]$. Without loss of generality, we also treat a truncated sequence as a vector of appropriate dimension. For an $N \times N$ Hermitian matrix Q, the ordered real eigenvalues are $\lambda_1(Q) \geq \lambda_2(Q) \geq \cdots \geq \lambda_N(Q)$. Further, $Q \geq 0$ (respectively, $Q > 0$) denotes that Q is positive semi-definite (respectively, positive definite). For a square matrix A, $\rho(A)$ denotes the spectral radius of A and diag(A) denotes the diagonal matrix containing the diagonal entries of A. $tr(.)$ is the trace operator. I_N denotes the $N \times N$ identity matrix, $\mathbf{1}_N = [1, 1, \ldots, 1]^T \in \mathbb{R}^N$ and, $\mathbf{0}_N = [0, 0, \ldots, 0]^T \in \mathbb{R}^N$. The $\mathcal{Q}$-function is $\mathcal{Q}(x) = \frac{1}{\sqrt{2\pi}} \int_x^\infty e^{-\frac{u^2}{2}} du$. Finally, $\mathbf{N}(0, \Sigma)$ denotes the standard normal distribution with mean 0 and covariance matrix Σ.

2. Problem formulation

In this section, we present our multi-agent cooperative optimisation problem and characterise its solution. Additionally, we describe a noise-adding mechanism that preserves the privacy of the agents' states over time.

2.1 Problem set-up

Consider a distributed system with a set of $N \geq 2$ agents denoted by $\mathcal{N} = \{1, 2, \ldots, N\}$. The agents collectively aim to minimise a common objective that is given by

$$\mathbf{P}: \qquad \min_x \quad J_{co}(x) = \tfrac{1}{2}x^T Q x + r^T x + s, \qquad (1)$$

where the vector $x = [x_1, x_2, \ldots, x_N]^T \in \mathbb{R}^N$ denotes the states of all agents. Further, Q is a non-zero $N \times N$ real matrix, $r \in \mathbb{R}^N$ and $s \in \mathbb{R}$. Let q_{ij} and r_i denote the entries of Q and r, respectively. Note that the states of the agents are coupled with each other via the quadratic term $\tfrac{1}{2}x^T Q x$ in the cost function. Specifically, we say that the agents i and j are uncoupled if both q_{ij} and q_{ji} are zero, and that they are coupled otherwise. Let $\mathcal{N}_i$ denote the *neighbour set* or the set of agents whose states are coupled to the state of agent i, and let $N_i = |\mathcal{N}_i|$. We place the following assumptions on the cost function $J_{co}(x)$:

(A.1) Q is symmetric and positive semi-definite. Further, if 0 is an eigenvalue of Q, then: (1) its algebraic multiplicity is 1, and (2) $r = \mathbf{0}_N$ in Equation (1).

(A.2) Each row(or column) of $Q - \text{diag}(Q)$ has at least one non-zero entry.

Assumption **A.2** implies that there is no uncoupled agent in the system, that is, $N_i \neq 0$ for each $i \in \mathcal{N}$. This assumption is not restrictive because a system with n uncoupled agents can be studied via a reduced system

with $N - n$ coupled agents. Assumption **A.1** implies that the minimisation problem **P** is convex and admits a finite (but not necessarily unique) optimum. Let the set of all the optimum solutions of **P** be denoted by $\mathcal{X}^*$. An optimum $x^* \in \mathcal{X}^*$ can be achieved by the agents with a distributed, iterative gradient descent algorithm. In such an algorithm, the update rule of agent i is

$$x_i(k+1) = x_i(k) - \gamma_1 \frac{\partial}{\partial x_i} J_{co}(x(k)) = x_i(k)$$
$$- \gamma_1 \left(q_{ii} x_i(k) + \sum_{j \in \mathcal{N}_i} q_{ij} x_j(k) + r_i \right), \quad (2)$$

where $\gamma_1 > 0$ is the step size and $x_i(0)$ is the initial state of agent i. As evident from the above iteration, agent i requires state information $x_{\mathcal{N}_i}$ from all its neighbours for its own state update. We assume that the agents can communicate their state information to each other without any distortion. The gradient descent algorithm for all agents can be collectively represented as

$$\mathbf{S_1}: \quad \begin{aligned} x(k+1) &= x(k) - \gamma_1(Qx(k) + r) \\ &= (I_N - \gamma_1 Q)x(k) - \gamma_1 r \triangleq A_1 x(k) + b_1, \end{aligned}$$
$$(3)$$

where $A_1 = I_N - \gamma_1 Q$, $b_1 = -\gamma_1 r$ and initial state $x(0) = [x_1(0), x_2(0), \ldots, x_N(0)]^T$. Since the cost gradient is linear, algorithm $\mathbf{S_1}$ can be represented as a discrete-time invariant linear system. Next, we derive the condition under which the steady-state solution of $\mathbf{S_1}$ exists and show that the steady-state solution is the optimum of problem **P**.

Lemma 2.1 (Convergence of algorithm $\mathbf{S_1}$): *Let $\gamma_1 < 2\rho(Q)^{-1}$. Then, the algorithm $\mathbf{S_1}$ in Equation (3) converges asymptotically to the optimum of P, that is $\lim_{k \to \infty} x(k) = x^*$ for an $x^* \in \mathcal{X}^*$.*

Proof: First, assume $Q > 0$. For $i = 1, 2, \ldots, N$ we have $0 < \gamma_1 \lambda_i(Q) \leq \gamma_1 \lambda_1(Q) < 2$. Since $\lambda_i(A_1) = 1 - \gamma_1 \lambda_i(Q)$, the above condition is equivalent to $-1 < \lambda_i(A_1) < 1$. Thus, all eigenvalues of A_1 lie inside the unit circle and a steady-state solution of Equation (3) is achieved. Assume now that Q has a 0 eigenvalue. Then, by assumption **A.1**, $b_1 = 0$ and A_1 has a single eigenvalue at 1 and all other eigenvalues lie inside the unit circle. Thus, the linear system in Equation (3) is marginally stable and a finite steady-state solution is achieved. Further, it can be easily observed that the steady-state solution of Equation (3) satisfies the first-order optimality condition $\nabla J_{co}(x) = Qx + r = \mathbf{0}_N$ of problem **P**. Thus, it minimises $J_{co}(x)$ and is an optimum of **P**. ∎

Let the steady-state of Equation (3) be denoted by m_1. Then,

$$m_1 = A_1 m_1 + b_1, \quad (4)$$

and the optimum cost achieved by the agents is given by

$$J_{co}^* \triangleq [J_{co}(x)]_{\mathbf{S_1}} = \frac{1}{2} m_1^T Q m_1 + r^T m_1 + s, \quad (5)$$

where the notation $[.]_{\mathbf{S_1}}$ denotes the cost achieved by the steady-state of algorithm $\mathbf{S_1}$.

Remark 2.1 (Examples): A number of problems fit into our quadratic cost framework, including consensus and formation control. We will discuss the consensus example in detail in Section 4. We also study the 1D CVT problem in which the agents implement a linear algorithm to minimise a convex cubic cost. This shows that our framework can be extended to problems involving convex non-quadratic costs that can be optimised by a linear algorithm.

2.2 Privacy mechanism

In the cooperative algorithm $\mathbf{S_1}$, the agents update their state upon communicating the state information with their neighbours. Thus, algorithm $\mathbf{S_1}$ is not private and in fact, an intruder may reconstruct the state trajectories of the agents with access to only a few messages communicated by the agents. To ensure privacy, we consider the DP mechanism that protects the state of the agents over time, where each agent adds an artificial random noise to its state before communicating it with other agents. The noise ensures that '*any two different instances*' of the communicated state trajectories are '*statistically not very different*', which prevents the intruder from accurately obtaining the actual state information of the agents; thus, maintaining their privacy. For the motivation and more information on DP, interested readers are referred to Dwork (2006) and Dwork (2011).

Remark 2.2 (Privacy of state trajectory): Note that different instances of the state trajectory arise from different initial states $x(0)$. However, in addition to the initial state, agents wish to keep their positions/velocities private at all times because accurate state information at any time instant can potentially reveal the complete future state trajectory.

The noisy DP mechanism can be written as

$$\mathcal{M}: \quad \tilde{x}_i(k) = x_i(k) + n_i(k), \quad (6)$$

where $\tilde{x}_i(k)$ denotes the state communicated to the neighbours of agent i and $n_i(k)$ is the random privacy noise. Let

$n(k) = [n_1(k), n_2(k), \ldots, n_N(k)]^T$. We adopt the DP framework developed in Le Ny and Pappas (2014) to design the noise that ensures privacy of the state trajectories. We begin with the definition of adjacency.

Definition 2.1 (Adjacency): Given a finite $\beta \geq 0$, two state trajectories $x[0:\infty]$ and $x'[0:\infty]$ are β-adjacent (denoted by $adj(\beta)$) if

$$\|x[0:\infty] - x'[0:\infty]\| \leq \beta. \qquad (7)$$

It should be noted that in the classic definitions of DP for static databases (Dwork, 2006) and for dynamical systems (Le Ny & Pappas, 2014), adjacency is defined with respect to the change of trajectory of one agent only, while keeping the trajectories of other agents unchanged. In contrast, our definition of adjacency allows simultaneous changes in the trajectories of one or more agents.

Remark 2.3 (Common steady-state value): The adjacency definition in Equation (7) implicitly requires that the two instances of the state trajectories (resulting from two different initial conditions) vary only for transient periods and have a common steady-state value. This holds true if $Q > 0$, since it is easy to observe (see (4)) that the steady-state value does not depend on the initial condition $x(0)$. However, when $Q \geq 0$ with a single eigenvalue at 0 (see **A.1**), then the steady- state value might depend on the initial condition. Let $\mathcal{X}_0^m$ denote the set of all initial conditions that result in a steady-state value of m. Then, the privacy mechanism guarantees DP only among those trajectories that result from initial conditions contained in the set $\mathcal{X}_0^m$.

Let $\tilde{x}[0:\infty]$ and $\tilde{x}'[0:\infty]$ denote the corresponding noisy communicated state trajectories. Note that $\tilde{x}[0:T] \in \mathbb{R}^{N(T+1)}$ and let $\mathcal{R}^{N(T+1)}$ denote the $\sigma - algebra$ generated by it. Next, we provide the definition of DP.

Definition 2.2 (Differential privacy): The mechanism $\mathcal{M}$ in Equation (6) is (ϵ, δ)-differentially private if for any two β-adjacent trajectories $x[0:\infty]$ and $x'[0:\infty]$ and for all $S \in \mathcal{R}^{N(T+1)}$ and for all $T \geq 0$ it holds

$$\mathbb{P}[\tilde{x}[0:T] \in S] \leq e^{\epsilon}\mathbb{P}[\tilde{x}'[0:T] \in S] + \delta, \qquad (8)$$

where $\epsilon > 0$ and $0 < \delta < 0.5$ are privacy parameters.

Definition 2.2 implies that for any two beta adjacent trajectories, the statistics of the corresponding noisy communicated state trajectories differ only within a multiplicative factor of e^{ϵ} and an additive factor of δ. A standard way to guarantee DP is to choose an i.i.d. Gaussian noise and scale its variance according to the adjacency parameter β, as stated in the next lemma.

Lemma 2.2 (Ensuring differential privacy): *The mechanism $\mathcal{M}$ in Equation (6) is (ϵ, δ)-differentially private if $n(k)$ is white Gaussian noise with distribution $n(k) \sim \mathbf{N}(0, \sigma^2 I_N)$, where*

$$\sigma \geq \frac{\beta}{2\epsilon}(K + \sqrt{(K^2 + 2\epsilon)}), \text{ and } K = \mathcal{Q}^{-1}(\delta).$$

Proof: See Theorem 3 in Le Ny and Pappas (2014). Since the quantity that needs to be protected and that is communicated is same (i.e. the state trajectory), the sensitivity is trivially upper bounded by the adjacency parameter β. ∎

In Lemma 2.2, the relation between σ and the privacy parameters (ϵ, δ) implies that the noise variance is a monotonically decreasing function of ϵ and δ. Also, note from the definition of DP in Equation (8) that a smaller value of ϵ and δ implies larger privacy of the agents. Thus, the noise variance σ can be treated as synonymous to the privacy level of the system, and for the ease of presentation, we present our results directly in terms of noise level σ (instead of the privacy parameters ϵ and δ).

In the presence of privacy noise, the evolution of the algorithm $\mathbf{S_1}$ in Equation (3) is modified as

$$\mathbf{S_1^{priv}} : x(k+1) = A_1 x(k) + b_1 + H_1 n(k), \qquad (9)$$

where $H_1 \triangleq A_1 - \text{diag}(A_1)$ is obtained by replacing the diagonal elements of A_1 with zero entries, since only non-diagonal entries in A_1 represent coupling between the agents. Note that $H_1 = -\gamma_1 \tilde{Q}$ where $\tilde{Q} = Q - \text{diag}(Q)$.

2.3 Performance degradation due to the privacy mechanism

The noise introduced by the privacy mechanism makes the states of the agents private. However, it also adversely affects the system performance. Due to the stochastic nature of algorithm $\mathbf{S_1^{priv}}$, we analyse the system performance by calculating the expected cost achieved by the agents in the presence of noise. The algorithm $\mathbf{S_1^{priv}}$ in Equation (9) can be viewed as a linear system driven by a constant input and Gaussian privacy noise. Thus, the state of the agents at each time instant has a normal distribution, denoted by $x(k) \sim \mathbf{N}(m_1(k), P_1(k))$. The evolution of the mean and the covariance of the states of the agents is given by

$$m_1(k+1) = A_1 m_1(k) + b_1, \quad \text{and} \qquad (10)$$
$$P_1(k+1) = A_1 P_1(k) A_1^T + \sigma^2 H_1 H_1^T, \qquad (11)$$

with $m_1(0) = x(0)$ and $P_1(0) = 0$. If A_1 is stable (i.e. $Q > 0$ in Equation (1)), then the mean and covariance reach

a finite steady-state value, denoted by m_1 and P_1, respectively. Note that we have overloaded the notation of m_1 with the noiseless case presented in Section 2.1 since the steady-state solutions of Equations (3) and (10) are the same. Thus, the mean m_1 satisfies Equation (4) and the covariance P_1 satisfies the following Lyapunov equation:

$$P_1 = A_1 P_1 A_1^T + \sigma^2 H_1 H_1^T. \qquad (12)$$

If A_1 is marginally stable (that is, Q in Equation (1) has a single eigenvalue at 0, see assumption **A.1**), then m_1 exists and is finite. However , the covariance P_1 may become unbounded, and the system becomes unstable in the stochastic sense. We now present the performance result.

Theorem 2.1 (Performance in the presence of privacy noise): *Assume $Q > 0$. At steady- state, the expected cost achieved by the agents implementing the algorithm $\mathbf{S_1^{priv}}$ in Equation (9) is given by*

$$J_{co}^*(\sigma) \triangleq \mathbb{E}[J_{co}(x)]_{\mathbf{S_1^{priv}}} = \frac{1}{2} tr(QP_1) + \frac{1}{2} m_1^T Q m_1 + r^T m_1 + s. \qquad (13)$$

where the expectation $\mathbb{E}[.]$ is taken w.r.t the privacy noise and P_1 depends on σ.

Proof: Since $Q > 0$, its Cholesky decomposition exists, denoted by $Q = L^T L$. Further, let x denote the random steady-state and let $y = Lx$. If $x \sim \mathbf{N}(m_1, P_1)$, then $y \sim \mathbf{N}(Lm_1, LP_1L^T)$. We have

$$\begin{aligned}
\mathbb{E}[J_{co}(x)]_{\mathbf{S_1^{priv}}} &= \mathbb{E}\left[\frac{1}{2}x^T Q x + r^T x + s\right] \\
&= \frac{1}{2}\mathbb{E}[y^T y] + r^T m_1 + s \\
&= \frac{1}{2}tr(\mathbb{E}[yy^T]) + r^T m_1 + s \\
&= \frac{1}{2}tr(LP_1L^T + Lm_1(Lm_1)^T) + r^T m_1 + s \\
&= \frac{1}{2}(tr(QP_1) + m_1^T Q m_1) + r^T m_1 + s,
\end{aligned}$$

where we have used the fact that $tr(.)$ is a linear and invariant under cyclic permutations. ∎

The performance degradation due to the privacy noise is obtained by comparing Equations (5) and (13), and is given by

$$J_{co}^*(\sigma) - J_{co}^* = \frac{1}{2}tr(QP_1),$$

which increases with the noise level σ. Because the agents share noisy state information, full cooperation and use of the distorted information for the algorithm updates affect the agents' performance. In the other extreme case, if the agents forgo cooperation, then they will be completely private, since no state information will be exchanged among them, but will probably not achieve the optimum of problem **P**. Thus, a mechanism is needed for the agents to adapt their level of cooperation to maximise their performance in the presence of privacy noise. In the next section, we define a notion of *cooperative level*, and present modified optimisation algorithms that incorporate the cooperation level as a parameter.

3. Cooperation level in multi-agent systems

In this section, we introduce and motivate our notion of cooperation level in private multi-agent systems. We calculate the expected cost achieved by the agents for a particular level of cooperation and privacy noise and use it to characterise the optimum cooperation level.

3.1 A notion of cooperation level

Agents cooperate to implement algorithm $\mathbf{S_1}$. However, as discussed above, full cooperation may not be optimal if agents also want to preserve privacy. To formalise this, we introduce a cooperation parameter in the algorithm $\mathbf{S_1}$. In many scenarios, in addition to minimising the system cost J_{co}, the agents also have individual goals for which no cooperation is required. As explained in the Introduction, such conflicting goals are ubiquitous in optimisation and game theory. We formalise the individual agent goals by the following cost function:

$$J_{nco}(x) = \frac{1}{2}x^T \bar{Q} x + \bar{r}^T x + \bar{s}, \qquad (14)$$

and assume that

(A.3) $\bar{Q}$ is diagonal and positive definite.

Assumption **A.3** implies that the states of all the agents are decoupled in $J_{nco}(x)$. As a result, no cooperation is required to minimise the decoupled cost function $J_{nco}(x)$.

We utilise these individual agent goals to introduce *cooperation level* in our framework. The costs $J_{co}(x)$ and $J_{nco}(x)$ represent two extremes on the cooperation scale. To minimise the former, full cooperation is necessary among the agents, while no cooperation is required for the latter. When the agents wish to keep private, it is prudent for them to give more weight to their individual goals as compared to the system goal. Following this reasoning and to capture the intermediate cooperation behaviour, we consider a new cost function which is precisely the convex combination of $J_{co}(x)$ and $J_{nco}(x)$:

$$J_\alpha(x) = \alpha J_{co}(x) + (1 - \alpha)J_{nco}(x), \qquad (15)$$

where the parameter $\alpha \in [0, 1]$ is the agents' *cooperation level*. Note that the new cost $J_\alpha(x)$ is convex due to convexity of $J_{co}(x)$ and $J_{nco}(x)$. The gradient descent algorithm that minimises J_α inherently introduces the cooperation level in our framework and in the presence of privacy noise can be written as

$$\mathbf{S}_\alpha^{\mathbf{priv}}: \quad \begin{aligned} x(k+1) &= x(k) - \gamma\,(\alpha\,(Qx(k) + r) \\ &\quad + (1-\alpha)(\bar{Q}x(k) + \bar{r})) + H_\alpha n(k) \\ &\triangleq A_\alpha x(k) + b_\alpha + H_\alpha n(k), \end{aligned} \quad (16)$$

where

$$\begin{aligned} Q_\alpha &= \alpha Q + (1-\alpha)\bar{Q}, \quad A_\alpha = I_N - \gamma Q_\alpha, \\ b_\alpha &= -\gamma r_\alpha, \quad r_\alpha = \alpha r + (1-\alpha)\bar{r}, \\ H_\alpha &\triangleq A_\alpha - \mathrm{diag}(A_\alpha) = -\gamma\alpha(Q - \mathrm{diag}(Q)), \end{aligned}$$

and $\gamma > 0$ is the step size.

Remark 3.1 (Auxiliary cost function): Note that the decoupled cost function J_{nco}, the new cost function J_α and its minimising algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ merely act as a means to introduce the cooperation level in our problem. The performance of the agents is measured in terms of the expected value of global cost $J_{co}(x)$ achieved when the agents follow algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$.

By varying the cooperation level in $\mathbf{S}_\alpha^{\mathbf{priv}}$, the agents can achieve a range of private solutions of **P**. In practice, agents should select a cooperation level that maximises the system performance, as we will show in the next subsection. Note that agents are still required to exchange their state information for all values of the cooperation level $0 < \alpha \leq 1$, and that the cooperation level determines the weight given by an agent to the information coming from its neighbours.

Remark 3.2 (Alternative approaches for variable cooperation level): Different methods exist to capture the notion of cooperation level. For instance, agents may filter the exchanged measurements to reduce the effect of privacy noise on the performance, and use the filter weights to measure their cooperation level. However, our formulation modulates cooperation in a natural and explicit way by balancing the individual and global goals of the agents, and it allows us to directly characterise critical trade-offs between privacy and performance in multi-agent systems.

Remark 3.3 (Selection of the decoupled cost): There may be scenarios in which the agents do not have individual goals. In such cases, we can construct an artificial decoupled cost J_{nco} to capture the non-cooperation extreme. Several choices of the matrix $\bar{Q}$ are possible. For

instance, $\bar{Q}$ can be chosen to consist of the diagonal elements of Q, or it can be an arbitrary matrix that satisfies assumption **A.3** (see examples in Section 4). Thus, our framework is applicable for a wide variety of multi-agent optimisation problems. The selection of a decoupled cost that optimises the agents performance is left as the subject of future research.

3.2 Performance analysis with privacy and cooperation

We now analytically characterise the expected cost for a given privacy and cooperation level. Note that due to assumptions **A.1** and **A.3**, both Q_α and A_α are symmetric. Moreover, since the privacy noise has a normal distribution, the state $x(k)$ in algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ is also normal. Let the steady-state mean and covariance of $x(k)$ in algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ be denoted by m_α and P_α, respectively. Next, we present conditions under which the steady-state mean and covariance exist. Note that Lemma 2.1 presents such condition for $\alpha = 1$.

Lemma 3.1 (Convergence of $\mathbf{S}_\alpha^{\mathbf{priv}}$ for $\alpha \neq 1$): *Let $\alpha \neq 1$. Then, the steady-state mean and covariance of algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ exist if*

$$\gamma < \frac{2}{\max\{\rho(Q), \rho(\bar{Q})\}}. \quad (17)$$

Proof: The proof is similar to that of Lemma 2.1 by using the following facts: (i) $Q_\alpha > 0$ for $\alpha \neq 1$, and (ii) from Weyl's inequality (Tao, 2012), $\rho(Q_\alpha) \leq \max\{\rho(Q), \rho(\bar{Q})\}$. ∎

Analogous to Equations (4) and (12), the steady-state mean and covariance of $\mathbf{S}_\alpha^{\mathbf{priv}}$ satisfy

$$\begin{aligned} m_\alpha &= A_\alpha m_\alpha + b_\alpha \\ \Rightarrow 0 &= Q_\alpha m_\alpha + r_\alpha \quad \text{and,} \quad (18) \\ P_\alpha &= A_\alpha P_\alpha A_\alpha^T + \sigma^2 H_\alpha H_\alpha^T. \quad (19) \end{aligned}$$

A closed-form expression of P_α can be written as

$$P_\alpha = \sigma^2 \gamma^2 \alpha^2 \sum_{k=0}^{\infty} A_\alpha^k (Q - \mathrm{diag}(Q))^2 (A_\alpha^T)^k. \quad (20)$$

Next, we characterise the performance of the agents when they follow the algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$. The performance is measured by the expected value of the global cost $J_{co}(x)$. Similarly to Equation (13), the cost achieved by algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$

is

$$J(\alpha, \sigma) \triangleq \mathbb{E}[J_{co}(x)]_{\mathbf{S}_\alpha^{\mathrm{priv}}} = J_{\mathrm{priv}}(\alpha, \sigma) + J_{ico}(\alpha) \quad \text{where,}$$

(21)

$$J_{\mathrm{priv}}(\alpha, \sigma) \triangleq \frac{1}{2}tr(QP_\alpha) \quad \text{and,}$$

(22)

$$J_{ico}(\alpha) \triangleq \frac{1}{2}m_\alpha^T Q m_\alpha + r^T m_\alpha + s.$$

(23)

Note that $J_{ico}(\alpha)$ represents the cost achieved by the agents for any *intermediate cooperation* level α in the absence of privacy noise. Further, the cost term $J_{\mathrm{priv}}(\alpha, \sigma)$ quantifies the effect of the privacy noise at a given cooperation level since the covariance P_α depends on noise level σ. However, we omit that dependence in the notation for the ease of presentation. Moreover, P_α also depends on the step size γ. We do not analyse this dependence because γ dictates the number of iterations for algorithm $\mathbf{S}_\alpha^{\mathrm{priv}}$ to converge, which is not the primary issue addressed in this paper. Note that the optimum for **P** is achieved only when $\alpha = 1$ and $\sigma = 0$ (that is, when the agents fully cooperate and no privacy noise is present). To clarify the notation, $J_{co}^* = J(1, 0)$, $J_{co}^*(\sigma) = J(1, \sigma)$ and $J_{ico}(\alpha) = J(\alpha, 0)$. Also note that the functions J, J_{priv} and J_{ico} are continuous functions in their respective variables.

Intuitively, in the absence of privacy noise, the performance should increase as the agents cooperate more and should equal the best performance when they cooperate fully ($\alpha = 1$). The following lemma proves this fact and justifies our definition of cooperation level.

Lemma 3.2 (Performance without privacy): *The cost $J_{ico}(\alpha)$ in Equation (23) is monotonically decreasing for $\alpha \in [0, 1]$.*

Proof: By differentiating $J_{ico}(\alpha)$ with respect to α, we obtain

$$J_{ico}'(\alpha) = (m_\alpha^T Q + r^T)m_\alpha'.$$

(24)

For $\alpha = 1$, from Equation (18) we have, $Qm_1 + r = 0$. Thus, $J_{ico}'(1) = 0$.

For $\alpha \in [0, 1)$, $Q_\alpha > 0$. Thus, Q_α is invertible, and $Q_\alpha^{-1} > 0$. Differentiating Equation (18), we get

$$(Q - \bar{Q})m_\alpha + Q_\alpha m_\alpha' + r - \bar{r} = 0,$$

(25)

$$\overset{(a)}{\Rightarrow} m_\alpha' = -\frac{Q_\alpha^{-1}(Qm_\alpha + r)}{1 - \alpha},$$

(26)

where (a) follows from Equation (18). Thus, we have

$$J_{ico}'(\alpha) = -\frac{1}{1-\alpha}(Qm_\alpha + r)^T Q_\alpha^{-1}(Qm_\alpha + r).$$

and the derivative is non-positive, which completes the proof. ■

Lemma 3.2 implies that, in the absence of privacy noise, it is beneficial for the agents to cooperate fully. Instead, in the presence of privacy noise, agents can achieve a range of private solutions of **P** by varying the cooperation level. In practice, agents should select a cooperation level that minimises the cost $J(\alpha, \sigma)$. The optimum cooperation level for the agents for a given level of privacy noise σ is characterised as

$$\alpha^*(\sigma) = \arg \min_\alpha J(\alpha, \sigma).$$

(27)

Remark 3.4 (Finding the optimum cooperation level): The optimum cooperation level $\alpha^*(\sigma)$ can be approximated numerically by discretising the interval $[0, 1]$ for α, and evaluating the cost $J(\alpha, \sigma)$ at each point.

Next, we show that under some conditions on cost functions J_{priv} and J_{ico}, we can characterise the behaviour of $\alpha^*(\sigma)$.

Theorem 3.1 (Characterising the optimum cooperation level): *For all $\sigma > 0$, let $J_{\mathrm{priv}}(\alpha, \sigma)$ be strictly increasing for all $\alpha \in (0, 1]$. Further, let $J_{\mathrm{priv}}(\alpha, \sigma)$ and $J_{ico}(\alpha)$ be strictly convex for $\alpha \in [0, 1)$. Then, $\alpha^*(\sigma)$ is a monotonically decreasing function of σ.*

Proof: Let $'$ denote the derivative or partial derivative w.r.t. α. Using Equation (20), we have $J_{\mathrm{priv}}(\alpha, \sigma) = \sigma^2 f(\alpha)$, where $f(\alpha) \triangleq \frac{1}{2}\alpha^2 \gamma^2 tr[Q \sum_{k=0}^\infty A_\alpha^k (Q - \mathrm{diag}(Q))^2 (A_\alpha^T)^k]$. Since $J_{\mathrm{priv}}(\alpha, \sigma)$ is assumed to be strictly increasing in the theorem statement, $f(\alpha)$ is also strictly increasing for $\alpha \in (0, 1]$. Also, it can be readily observed that $f'(0) = 0$ and $f'(0) > 0$. From the proof of Lemma 3.2, we have $J_{ico}'(0) \leq 0$ and $J_{ico}'(1) = 0$. Thus, $J'(0, \sigma) \leq 0$ and $J'(1, \sigma) > 0$. Also, $J(\alpha, \sigma)$ is strictly convex for $\alpha \in [0, 1)$. Thus, $\alpha^*(\sigma) \in [0, 1)$ is unique and $J'(\alpha^*(\sigma), \sigma) = 0$.

We prove the theorem by contradiction. Suppose $\alpha^*(\sigma)$ is not monotonically decreasing function. Then, there exist $0 < \sigma_1 < \sigma_2$ such that $0 \leq \alpha^*(\sigma_1) < \alpha^*(\sigma_2) < 1$. Further,

$$J'(\alpha^*(\sigma_1), \sigma_1) = \sigma_1^2 f'(\alpha^*(\sigma_1)) + J_{ico}'(\alpha^*(\sigma_1)) = 0, \quad \text{and}$$
$$J'(\alpha^*(\sigma_2), \sigma_2) = \sigma_2^2 f'(\alpha^*(\sigma_2)) + J_{ico}'(\alpha^*(\sigma_2)) = 0.$$

Subtracting, we get

$$0 = \sigma_1^2 f'(\alpha^*(\sigma_1)) - \sigma_2^2 f'(\alpha^*(\sigma_2)) + J_{ico}'(\alpha^*(\sigma_1))$$
$$- J_{ico}'(\alpha^*(\sigma_2))$$
$$\overset{(a)}{\leq} \sigma_1^2 [f'(\alpha^*(\sigma_1)) - f'(\alpha^*(\sigma_2))] + J_{ico}'(\alpha^*(\sigma_1))$$
$$- J_{ico}'(\alpha^*(\sigma_2)) \overset{(b)}{<} 0,$$

where (*a*) follows since *f* is an increasing function and (*b*) follows from the strict convexity of *f* and J_{ico}. Thus, there is a contradiction and therefore, the theorem follows. ∎

Due to the convexity and increasing properties of the functions involved in Theorem 3.1, we are able to obtain a nice characterisation of the optimum cooperation level. This result implies that under the conditions given in Theorem 3.1, it is always beneficial for the agents to reduce their cooperation level if they want to increase their privacy level. It characterises an important and previously unidentified trade-off between privacy and cooperation in multi-agent systems. Next, we show how to design the artificial cost function J_{nco} (in cases where individual costs are not present) which guarantees that the conditions of Theorem 3.1 hold true.

Corollary 3.1 (Design of artificial cost function): *Assume that Q satisfies the following properties:*

   (i) $Q > 0$ and $\frac{\lambda_1(Q)}{\lambda_N(Q)} < 1.5$,
   (ii) $\text{diag}(Q) = \mu I_N$ for some $\mu > 0$.

Then, there exists a $\bar{Q} = \delta I_N$ with (iii) $\lambda_1(Q) < \delta < 1.5\lambda_N(Q)$ and (iv) $\gamma < \delta^{-1}$ such that $J_{\text{priv}}(\alpha, \sigma)$ is strictly increasing for $\alpha \in (0, 1]$, and $J_{ico}(\alpha)$ and $J_{\text{priv}}(\alpha, \sigma)$ are strictly convex for $\alpha \in [0, 1)$.

Proof: See Appendix. ∎

The above corollary guarantees that $\alpha^*(\sigma)$ is a monotonically decreasing function. It should be noticed, however, that the conditions presented in Theorem 3.1 and Corollary 3.1 are not necessary, and $\alpha^*(\sigma)$ may or may not exhibit similar behaviour if these conditions are not satisfied.

4. Consensus and Voronoi tessellation

In this section, we illustrate our results through two prototypical problems, namely the consensus and the 1D Voronoi tessellation problems.

4.1 Consensus with privacy and cooperation

Consensus algorithms are used by autonomous agents to agree on a common value in a distributed fashion. The algorithm involves sharing of state information among the agents. The iterations of the consensus algorithm in a discrete-time setting can be represented as (Olfati-Saber et al., 2007)

$$x(k+1) = (I_N - \gamma L)x(k), \quad (28)$$

where $L = [l_{ij}]$ is the Laplacian matrix of the graph representing the agents' interaction. We consider an undirected consensus graph for which the Laplacian is symmetric, positive semi-definite with positive diagonal entries and non-positive non-diagonal entries. For such a graph, $\mathbf{1}_N$ is an eigenvector of L associated with the eigenvalue 0, that is, $L\mathbf{1}_N = 0$ and $\lambda_N(L) = 0$. Further, we assume that the graph is connected, which is equivalent to $\lambda_{N-1}(L) > 0$. Then, the algorithm in Equation (28) asymptotically achieves average consensus, that is, $\lim_{k\to\infty} x(k) = \mu \mathbf{1}_N$, where $\mu = \frac{1}{N}\sum_{i=1}^{N} x_i(0)$.

The consensus algorithm in Equation (28) can be viewed as a gradient descent algorithm to minimise the following disagreement function (Olfati-Saber et al., 2007):

$$J_{co}(x) = \frac{1}{2}x^T L x = \frac{1}{4}\sum_{i=1}^{N}\sum_{j:j\in\mathcal{N}_i} -l_{ij}(x_i - x_j)^2. \quad (29)$$

Thus, the consensus problem fits into our framework (Problem (1) and solution (3)) with $Q = L$, $r = 0$, and $s = 0$. Further, it also satisfies assumptions **A.1** and **A.2**. To introduce the cooperation level, we select the following decoupled cost function:

$$J_{nco}(x) = \frac{1}{2N}x^T x - \frac{1}{N}a^T x + \frac{1}{2N}a^T a + b^2, \quad (30)$$

where $a \in \mathbb{R}^N$ and $b \in \mathbb{R}$. The optimum of $J_{nco}(x)$ is achieved at $x = a$ and the optimum cost is b^2. The non-cooperative cost J_{nco} signifies the fact that each agent is stubborn: i.e. it wants the opinions (states) of the other agents to converge to its own, without changing its own opinion. Consensus with stubborn (or persistent) agents is well studied and J_{nco} corresponds to the cost when all agents are fully stubborn (see Ghaderi & Srikant, 2013, Equation (1)). Since all agents are stubborn, they do not cooperate and evolve independently without using each other's state information.

Comparing the above decoupled cost function with Equation (14), we obtain $\bar{Q} = \frac{1}{N}I_N$, $\bar{r} = -\frac{1}{N}a$, and $\bar{s} = \frac{1}{2N}a^T a + b^2$. Thus,

$$Q_\alpha = \alpha L + \frac{1-\alpha}{N}I_N, \quad A_\alpha = I_N - \gamma Q_\alpha, \quad b_\alpha = \frac{\gamma(1-\alpha)}{N}a.$$

Further, the step size can be chosen according to Lemma 3.1 to guarantee convergence of the consensus algorithm. The resulting cost with privacy mechanism becomes

$$J(\alpha, \sigma) = \frac{1}{2}(tr(LP_\alpha) + m_\alpha^T L m_\alpha).$$
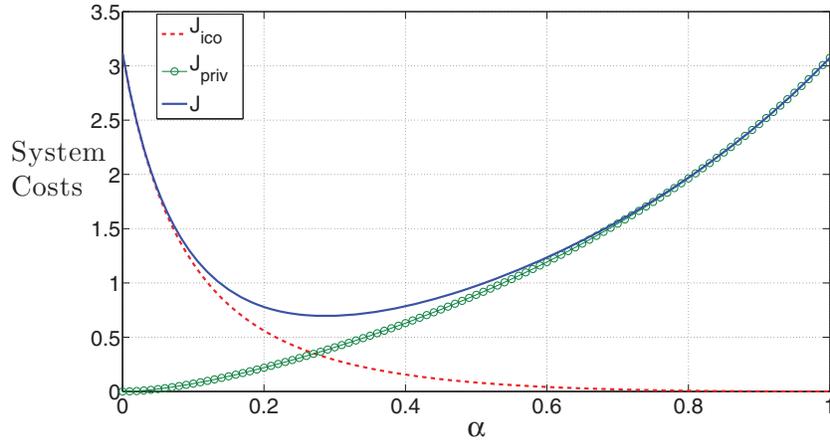
Figure 1. System costs as a function of cooperation level for privacy noise level $\sigma = 3$.

Note that, for $\alpha = 1$, A_1 becomes marginally stable and thus, the covariance matrix P_1 becomes unbounded resulting in instability (see discussion below (12)). Thus, the agents have to choose a cooperation level $0 \leq \alpha < 1$ for the cost to remain finite.

We now consider a specific consensus example with $N = 4$ agents and the following Laplacian matrix:

$$L = \begin{bmatrix} 0.8 & -0.14 & -0.15 & -0.51 \\ -0.14 & 1.4 & -0.85 & -.41 \\ -0.15 & -0.85 & 1.1 & -0.1 \\ -0.51 & -0.41 & -0.1 & 1.02 \end{bmatrix}.$$

Let $a = [1.5, 1, 3, 0]^T$ and $x(0) = [0.2, 0.6, 1.2, 2]^T$. Figure 1 shows the system costs J, J_{priv} and J_{ico} in Equations (21)–(23) as a function of α for $\sigma = 3$. We can observe that J_{ico}, which is the system cost in the absence of privacy noise, is monotonically decreasing (cf. Lemma 3.2). This validates our understanding that it is always beneficial for the agents to have full cooperation if they do not desire any privacy.

The effect of privacy noise is included in the cost J_{priv}. It is interesting to observe its behaviour at different cooperation levels. Note that the noise has only a marginal effect on the cost at smaller cooperation levels. In fact, when the agents do not cooperate ($\alpha = 0$), the noise does not affect the performance at all – which is natural because the agents do not share any information. In contrast, the effect of noise is significantly higher at larger cooperation levels, because the agents use the noisy states in their updates. Thus, the cost J_{priv} is a monotonically increasing function of α. The two cost curves J_{ico} and J_{priv} highlight the trade-off between having full cooperation vs. no cooperation. As evident from the resulting overall

cost curve J, an intermediate optimum cooperation level should be chosen to achieve best performance.

Figure 2 shows the overall cost J achieved as a function of the cooperation level for various levels of privacy noise. Observe that for each cooperation level, the cost increases with the noise level. These curves highlight that the optimum cooperation level changes with the privacy noise level which can be seen explicitly in Figure 3. Note that along with the fact that J_{priv} is an increasing function, both J_{priv} and J_{ico} are strictly convex. Thus, $\alpha^*(\sigma)$ is a monotonically decreasing function (cf. Theorem 3.1), which implies that it is always better for the agents to reduce their cooperation level if they desire to have a higher level of privacy. Finally, note that the consensus example does not satisfy the conditions in Corollary 3.1 (see discussion below Corollary 3.1).

4.2 Centroidal Voronoi tessellation

In this subsection, we show that our results and the intuition gained from the consensus problem hold even when some of our assumptions are not satisfied; thus suggesting that a trade-off between privacy and cooperation exists in a large class of distributed systems. We study a 1D CVT problem over the interval $\Omega = [0, 1]$. The goal of a CVT problem is to divide Ω into N regions denoted by $\{\Omega_i\}_{i \in \mathcal{N}}$, and find N points in Ω, denoted by $\{x_i\}_{i \in \mathcal{N}}$ such that (1) Ω_i is the Voronoi region[1] of x_i, and (1) x_i is the centroid of Ω_i. The CVT problem can be expressed as the following optimisation problem:

$$\min_{x} \quad J_{co}(x) = \int_0^1 \min_{i \in \mathcal{N}} (x_i - y)^2 dy. \tag{31}$$
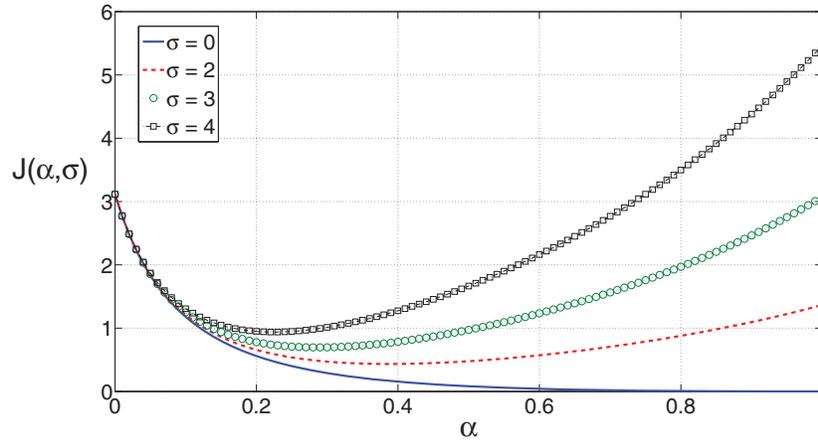
Figure 2. Consensus cost as a function of cooperation and privacy.

The 1D-CVT problem can be viewed as an optimal agent placement problem , and it has wide applications including data compression and scalar quantisation (Du, Faber, & Gunzburger, 1999).

Without loss of generality, we assume that $x_1 \leq x_2 \leq \cdots \leq x_N$. Then, the cost in Equation (31) can be written as

$$J_{co}(x) = \frac{1}{24} \sum_{i=1}^{N} \left[(x_i - x_{i-1})^3 + (x_{i+1} - x_i)^3 \right], \quad (32)$$

where $x_0 \triangleq -x_1$ and $x_{N+1} \triangleq 2 - x_N$ are dummy variables introduced for ease of analysis. It can be easily verified that $J_{co}(x)$ is convex.[2] Further, the ith component of its gradient can be written as

$$\frac{\partial J_{co}(x)}{\partial x_i} = \frac{1}{4} (2x_i - x_{i-1} - x_{i+1})(x_{i+1} - x_{i-1}). \quad (33)$$

By examining Equation (33), it follows that the optimum value x^* that minimises $J_{co}(x)$ also minimises a different cost function, denoted by $\tilde{J}_{co}(x)$, whose ith component of the gradient is

$$\frac{\partial \tilde{J}_{co}(x)}{\partial x_i} = \frac{1}{4} (2x_i - x_{i-1} - x_{i+1}). \quad (34)$$

Since $J_{co}(x)$ and $\tilde{J}_{co}(x)$ have the same optimum, we can use the gradient of either cost functions in the gradient descent algorithm for solving the CVT problem. However, the gradient of $J_{co}(x)$ is nonlinear whereas the gradient of the $\tilde{J}_{co}(x)$ is linear. We choose the linear gradient since it results in a linear gradient descent algorithm and fits into our framework (3). Using the individual components in Equation (34), the gradient can be written as

$$\nabla \tilde{J}_{co}(x) = Qx + r, \quad (35)$$
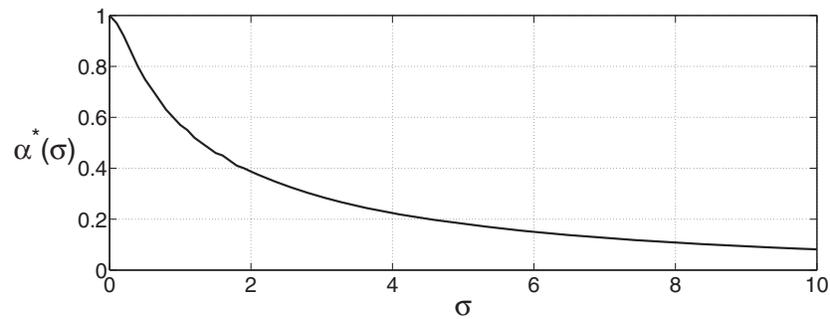
Figure 3. Variation of the optimum cooperation level with noise.

where $r = [0, 0, \ldots, 0, -0.5]^T$, and Q is the following tri-diagonal matrix:

$$Q = \frac{1}{4} \begin{bmatrix} 3 & -1 & 0 & 0 & 0 & \cdots & 0 \\ -1 & 2 & -1 & 0 & 0 & \cdots & 0 \\ 0 & -1 & 2 & -1 & 0 & \cdots & 0 \\ \vdots & & & \vdots & & & \vdots \\ 0 & 0 & \cdots & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & \cdots & 0 & -1 & 3 \end{bmatrix}.$$

Lemma 4.1 (Properties of Q): Q is positive definite and $\rho(Q) = 1$.

Proof: From Yueh (2005), Theorem 5, we get

$$\lambda_i(Q) = \frac{1}{2}\left(1 + \cos\frac{(i-1)\pi}{N}\right) \quad \text{for} \quad i = 1, 2, \ldots, N.$$

Thus, $\lambda_1(Q) = 1$ and $\lambda_N(Q) > 0$ and Lemma follows. ∎

The new cost function can then be written as $\tilde{J}_{co}(x) = \frac{1}{2}x^T Q x + r^T x$ and the gradient descent to solve the CVT problem in Equation (31) becomes

$$\mathbf{S_{CVT}}: \quad x(k+1) = x(k) - \gamma_1(Qx(k) + r) \triangleq A_1 x(k) + b_1. \tag{36}$$

The algorithm $\mathbf{S_{CVT}}$ with $\gamma_1 = 1$ is the well-known Lloyd's Algorithm (Du et al., 1999) for solving CVT problems in 1D.

Remark 4.1 (Generalisation for non-quadratic cost functions): Although the CVT problem has a cubic cost function, its optimum can be achieved via a linear algorithm so that our framework is still applicable. This suggests that our framework may be applicable to general non-quadratic convex cost functions, provided that the optimum can be achieved via a linear algorithm.

To introduce the cooperation level, we consider the following quadratic cost:

$$J_{nco}(x) = \frac{1}{N}\sum_{i=1}^{N}\int_0^1 (x_i - y)^2 dy = \frac{1}{N}x^T x - \frac{1}{N}\mathbf{1}_N^T x + \frac{1}{3}. \tag{37}$$

Recall that the optimisation with the coupled cost in Equation (31) can be interpreted as an optimal coverage problem (Du et al., 1999) wherein N agents want to cover the interval $[0, 1]$ in an optimal cooperative manner. On the other hand, the decoupled cost in Equation (37) represents the case when each agent wants to cover the whole interval by itself, without cooperating with other agents. Thus, the decoupled cost in Equation (37) is the counterpart of the coupled cost in Equation (31).

Note that $J_{nco}(x)$ satisfies assumption **A.3** and $x = [0.5, 0.5, \ldots, 0.5]^T$ is the optimum of $J_{nco}(x)$. By comparing the above cost function with Equation (14), we obtain $\bar{Q} = \frac{2}{N}I_N$, $\bar{r} = -\frac{1}{N}\mathbf{1}_N$ and $\bar{s} = \frac{1}{3}$, and the corresponding values of $Q_\alpha, A_\alpha, b_\alpha$ and H_α can be calculated using Equation (16). Since Q_α is positive definite, the stability condition (17) for the CVT problem reduces to $\gamma < 2$. We next analyse the system cost and performance.

The cost in Equation (32) can be simplified as

$$J_{co}(x) = \frac{1}{3}x_1^3 + \frac{1}{12}\sum_{i=1}^{N-1}(x_{i+1} - x_i)^3 + \frac{1}{3}(1 - x_N)^3.$$

We make the linear transformation $z = Gx + g$ to obtain

$$\underbrace{\begin{bmatrix} x_1 \\ x_2 - x_1 \\ x_3 - x_2 \\ \vdots \\ x_N - x_{N-1} \\ 1 - x_N \end{bmatrix}}_{z} = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \cdots & 0 \\ -1 & 1 & 0 & \cdots & 0 \\ 0 & -1 & 1 & 0 & \cdots \\ & & \vdots & & \\ 0 & \cdots & 0 & -1 & 1 \\ 0 & 0 & \cdots & 0 & -1 \end{bmatrix}}_{G} x + \underbrace{\begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}}_{g}.$$

Let m_α and P_α denote the steady-state mean and covariance achieved by algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ in Equation (16) for the CVT problem. Further, let η_α and V_α denote the steady-state mean and covariance of z, and let η_i and v_{ij} denote the elements of η_α and V_α, respectively. Due to the linear transformation, we have $\eta_\alpha = Gm_\alpha + g$ and $V_\alpha = GP_\alpha G^T$. Recall that, for a scalar random variable with distribution $y \sim \mathbf{N}(v, \theta^2)$, it holds $\mathbb{E}[y^3] = v^3 + 3v\theta^2$. Thus, the expected cost for the CVT problem becomes

$$J(\alpha, \sigma) = \frac{1}{3}(\eta_1^3 + 3\eta_1 v_{11}) + \frac{1}{12}\sum_{i=2}^{N}[\eta_i^3 + 3\eta_i v_{ii}] + \frac{1}{3}(\eta_{N+1}^3 + 3\eta_{N+1}v_{N+1,N+1}). \tag{38}$$

Consider now an example with $N = 4$ agents and $\gamma = 1$. Figure 4 shows how the steady-state values achieved by the agents vary with the cooperation level, when they use algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ in Equation (16) in the absence of noise. Observe that, when $\alpha = 0$, the optimum of $J_{nco}(x)$ in Equation (37) is $x = [0.5, 0.5, 0.5, 0.5]^T$, so that the agents occupy the same location due to the lack of cooperation. When $\alpha = 1$, the agents cooperate completely and achieve the solution $x = [\frac{1}{8}, \frac{3}{8}, \frac{5}{8}, \frac{7}{8}]^T$, which is the optimum of $J_{co}(x)$ in Equation (31). Figure 4 shows the solution achieved by algorithm $\mathbf{S}_\alpha^{\mathbf{priv}}$ for the intermediate values of α.

Figure 5 shows the system cost as a function of the cooperation level for various values of σ. Similar to the consensus example, we observe that the cost is a convex
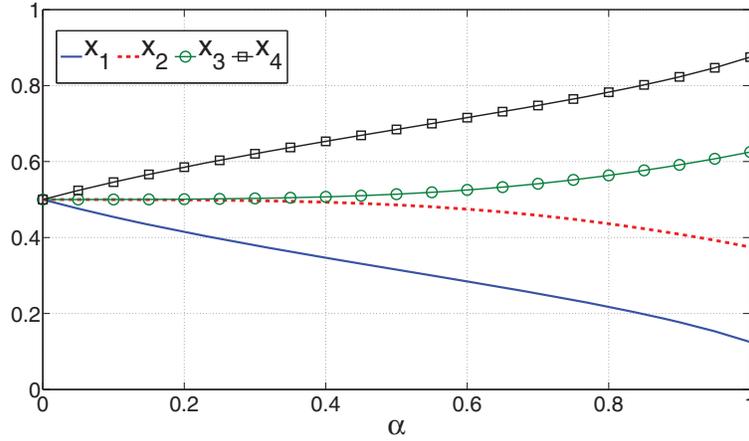
Figure 4. Steady-states achieved by S_α for the CVT problem in the absence of noise.

function of α and the optimum α is less than 1. Moreover, the cost increases when the noise level increases.

Analysis with two agents: The analysis of the cost function simplifies if we consider the simple, albeit useful, case of $N = 2$ agents. In this case, we can algebraically solve the steady-state Equations (18) and (19) to obtain

$$m_\alpha = \begin{bmatrix} \frac{1}{2} - \frac{\alpha}{4} & \frac{1}{2} + \frac{\alpha}{4} \end{bmatrix}, \text{ and } P_\alpha = \begin{bmatrix} p_1 & p_2 \\ p_2 & p_1 \end{bmatrix},$$

where $p_1 = \frac{\alpha^2 \sigma^2 (8 - \alpha^2)}{32(4 - \alpha^2)}$ and $p_2 = \frac{\alpha^4 \sigma^2}{32(4 - \alpha^2)}$. Thus,

$$J(\alpha, \sigma) = \frac{1}{48} + \frac{(1-\alpha)^2}{16} + \frac{\sigma^2 \alpha^2 (4 + \alpha)}{32(2 + \alpha)}. \quad (39)$$

It can be easily verified that this cost is monotonically increasing w.r.t. σ and is convex with respect to α. Consequently, an optimum α exists for each value of σ,

and it is given by

$$\alpha^*(\sigma) = \frac{\sqrt{(1 + 2\sigma^2)^2 + 8} - (1 + 2\sigma^2)}{2}. \quad (40)$$

As shown in Figure 6, the function $\alpha^*(\sigma)$ is monotonically decreasing. This indicates that it is best for the agents to cooperate fully when no noise is present, and to reduce their cooperation level when the privacy noise increases.

Remark 4.2 (Similarity between consensus and CVT results): By comparing Figures 5 and 6 in the CVT example (with cubic cost) with Figures 2 and 3 in the consensus example (with quadratic cost), we observe a similar pattern in the variation of the cost and the optimum cooperation level. This observation strengthens our belief that a similar trade-off should appear in problems with general non-quadratic convex cost function that can
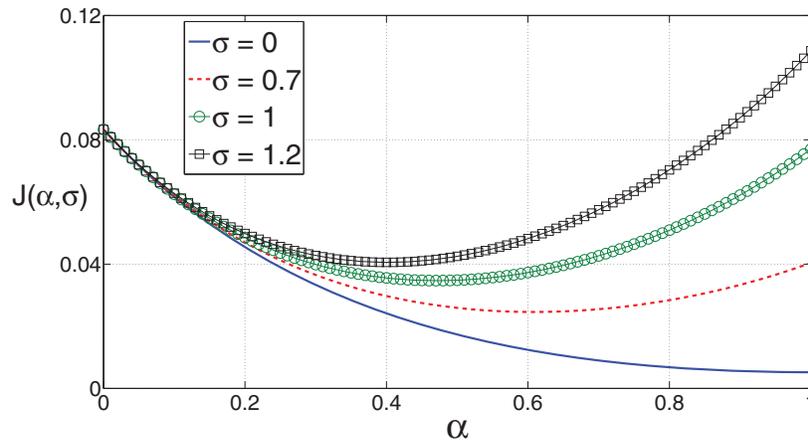
Figure 5. CVT cost as a function of cooperation level and noise level.
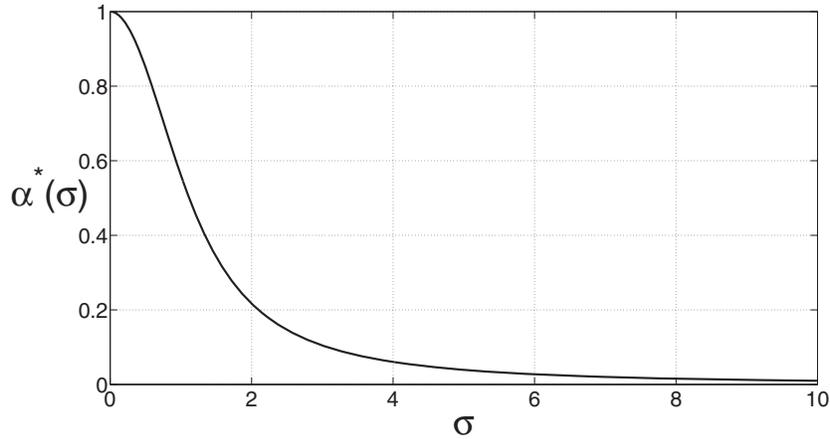
Figure 6. Variation of the optimum cooperation level with noise for CVT problem.

be minimised via linear iterations. This generalisation is left as a subject of future research.

5. Conclusion

This paper considers a multi-agent system where the agents cooperatively optimise a quadratic cost function while ensuring privacy of their states over time. First, we developed a noise-adding DP mechanism to protect the privacy of agents' states over time. Next, we characterised the system performance degradation due to the privacy noise and argued that the degradation occurs due to cooperation between the agents. Further, we developed a framework in which the agents can respond to the privacy noise present in the system by varying their cooperation level and studied the combined effect of privacy noise and cooperation level on the system performance. Using the performance characterisation, we next showed that there exists an optimum cooperation level that minimises the system cost and obtained conditions under which the optimum cooperation level is a decreasing function of privacy noise level. Finally, we studied two examples of consensus and Voronoi tessellations, and showed that they fit into our framework. The results obtained in the paper illustrate a trade-off between performance, privacy and cooperation, and suggest that, to optimise performance, agents should decrease their cooperation level if they want to increase the privacy level. Among directions of future research, it would be of interest to extend this work to arbitrary convex cost functions.

Notes

1. The Voronoi region for x_i is defined by $V_{x_i} = \{y : |y - x_i| \leq |y - x_j| \quad \forall \ j \neq i\}$.

2. For any $x \in \Omega^N, y \in \Omega^N$ and $0 < \lambda < 1, J_{co}(\lambda x + (1 - \lambda)y) \leq \lambda J_{co}(x) + (1 - \lambda)J_{co}(y)$.

Disclosure statement

No potential conflict of interest was reported by the authors.

Funding

References

Cortes, J., Martinez, S., Karatas, T., & Bullo, F. (2004). Coverage control for mobile sensing networks. *IEEE Transactions on Robotics and Automation, 20*(2), 243–255.

Du, Q., Faber, V., & Gunzburger, M. (1999). Centroidal Voronoi tessellations: Applications and algorithms. *Siam Review, 41*(4), 637–676.

Du, H., , Wen, G., Cheng, Y., He, Y., & Jia, R. (2016). Distributed finitetime cooperative control of multiple high order non-holonomic mobile robots. *IEEE Transactions on Neural Networks and Learning Systems,* Advance online publication. doi: 10.1109/TNNLS.2016.2610140.

Dwork, C. (2006). Differential privacy. *Proceedings of ICALP, 4052,* 1–12.

Dwork, C. (2011). A firm foundation for private data analysis. *Communications of the ACM, 54*(1), 86–95.

Ghaderi, J., & Srikant, R. (2013). Opinion dynamics in social networks: A local interaction game with stubborn agents. In *Proceedings of the 2013 American Control Conference*, (pp. 1982–1987). Washington, DC.

Hale, M., & Egerstedt, M. (2015). Cloud-enabled multi-agent optimization with constraints and differentially private states. Retrived from http://arxiv.org/abs/1507.04371.

Han, S., Topcu, U., & Pappas, G. J. (2014). Differentially private convex optimization with piecewise affine objectives. In *Proceedings of the 53rd IEEE Conference on Decision and Control*, (pp. 2160–2166). Los Angeles, CA.

Han, S., Topcu, U., & Pappas, G. J. (2017). Differentially private distributed constrained optimization. *IEEE Transactions on Automatic Control*, 62(1), 50–64.

Hsu, J., Roth, A., Roughgarden, T., & Ullman, J. (2014). Privately solving linear programs. *Automata, Languages, and Programming: ICALP, 14*, 612–624.

Huang, Z., Mitra, S., & Dullerud, G. (2012). Differentially private iterative synchronous consensus. In *Proceedings of the 2012 ACM Workshop on Privacy in the Electronic Society*. (pp. 81–90). New York, NY: ACM.

Huang, Z., Mitra, S., & Vaidya, N. (2015). Differentially private distributed optimization. In *Proceedings of the 2015 International Conference on Distributed Computing and Networking*. New York, NY: ACM.

Le Ny, J., & Pappas, G. J. (2014). Differentially private filtering. *IEEE Transactions on Automatic Control, 59*(2), 341–354.

Lindell, Y., & Pinkas, B. (2009). Secure multiparty computation for privacy-preserving data mining. *Journal of Privacy and Confidentiality, 1*(1), 59–98.

McDaniel, P., & McLaughlin, S. (2009). Security and privacy challenges in the smart grid. *IEEE Security and Privacy, 7*(3), 75–77.

Mo, Y., & Murray, R. M. (2017). Privacy preserving average consensus. *IEEE Transactions on Automatic Control*, 62(2), 753–765.

Nedic, A., & Ozdaglar, A. (2009). Distributed subgradient methods for multi-agent optimization. *IEEE Transactions on Automatic Control, 54*(1), 48–61.

Olfati-Saber, R. (2006). Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control, 51*(3), 401–420.

Olfati-Saber, R., Fax, J. A., & Murray, R. M. (2007). Consensus and cooperation in networked multi-agent systems. *Proceedings of the IEEE, 95*(2), 215–233.

Orlandi, C. (2011). Is multiparty computation any good in practice?. In *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, (pp. 5848–5851).Prague.

Raffard, R. L., Tomlin, C. J., & Boyd, S. P. (2004). Distributed optimization for cooperative agents: Application to formation flight. In *Proceedings of the 43rd IEEE Conference on Decision and Control*, (pp. 2453–2459). Nassau, Bahamas.

Rajagopal, R., & Wainwright, M. J. (2011). Network-based consensus averaging with general noisy channels. *IEEE Transactions on Signal Processing, 59*(1), 373–385.

Rosenthal, R. (1973). A class of games possessing pure-strategy nash equilibria. *International Journal of Game Theory, 2*(1), 65–67.

Tao, T. (2012). Topics in random matrix theory. *Graduate Studies in Mathematics, American Mathematical Society, 132*.

Terelius, H., Topcu, U., & Murray, R. M. (2011). Decentralized multi-agent optimization via dual decomposition. *Proceedings of the 18th IFAC World Congress, 44*(1), 11245–11251.

United States, D. o. E. (2010). Data access and privacy issues related to smart grid technologies (Technical report). United States Department of Energy, Retrieved from https://energy.gov/sites/prod/files/gcprod/documents/Broadband_Report_Data_Privacy_10_5.pdf.

Xiao, L., Boyd, S., & Kim, S. J. (2007). Distributed average consensus with least-mean-square deviation. *Journal of Parallel and Distributed Computing, 67*(1), 33–46.

Yueh, W. C. (2005). Eigenvalues of several tridiagonal matrices. *Applied Mathematics E-Notes, 5*, 66–74.

Appendix. *Proof of Corollary 3.1*

Proof: Let $'$ denote the derivative or partial derivative w.r.t. α. First, we derive conditions under which the cost term $J_{ico}(\alpha)$ is convex w.r.t. α. By differentiating Equation (24) with respect to α we obtain

$$J''_{ico}(\alpha) = (m'_\alpha)^T Q m'_\alpha + (Q m_\alpha + r)^T m''_\alpha.$$

From the proof of Lemma 3.2, for $\alpha = 1$, we have $Q m_1 + r = 0$ and $m'_1 = 0$. Thus, $J''_{ico}(1) = 0$. Now let $\alpha \in [0, 1)$. By differentiating Equation (25), we get

$$2(Q - \bar{Q})m'_\alpha + Q_\alpha m''_\alpha = 0,$$
$$\overset{(a)}{\Rightarrow} m''_\alpha = \tfrac{2}{1-\alpha} Q_\alpha^{-1}(Q - \bar{Q})Q_\alpha^{-1}(Q m_\alpha + r),$$

where (a) follows from Equation (26). Substituting the derivatives m'_α and m''_α, we get

$$J''_{ico}(\alpha) = \frac{1}{1-\alpha}(Q m_\alpha + r)^T Q_\alpha^{-1}$$
$$\times \left(3Q - 2\bar{Q} + \frac{\alpha}{1-\alpha}Q\right) Q_\alpha^{-1}(Q m_\alpha + r).$$

Condition (iii) in the corollary guarantees that $3Q - 2\bar{Q} > 0$ and thus $J''_{ico}(\alpha) > 0$ for $\alpha \in (0, 1]$.

Next, we derive conditions under which the cost term $J_{priv}(\alpha, \sigma)$ is convex w.r.t. α. Recalling Equation (16), let $A_\alpha = \alpha A + B$, where $A \triangleq \gamma(\bar{Q} - Q)$ and $B \triangleq I_N - \gamma\bar{Q} = (1 - \gamma\delta)I_N$. Further, let $H_\alpha = -\gamma\alpha\tilde{Q}$ where $\tilde{Q} \triangleq Q - \text{diag}(Q)$. Differentiating Equation (19) and substituting the above expressions, we get

$$P'_\alpha = A_\alpha P'_\alpha A_\alpha + \underbrace{A_\alpha P_\alpha B^T + B P_\alpha A^T + 2\alpha A P_\alpha A^T + 2\sigma^2 \gamma^2 \alpha \tilde{Q}^2}_{W}.$$
(A1)

Note that due to (iii) and (iv), $A > 0$ and $B > 0$. Further, we have the following facts (a) $Q A_\alpha = A_\alpha Q$ and (b) $Q\tilde{Q} = \tilde{Q}Q$ (by (ii)). Using (a), (b) and Equation (20), it can be easily observed that $A P_\alpha B^T = B P_\alpha A^T > 0$. Thus, $W > 0$ and (A1) resembles a Lyapunov equation. Hence, we conclude that $P'_\alpha > 0$.

Taking derivative of (A1), we get

$$P''_\alpha = A_\alpha P''_\alpha A_\alpha$$
$$+ \underbrace{2A P'_\alpha B^T + 2B P'_\alpha A^T + 2A(P_\alpha + 2\alpha P'_\alpha)A^T + 2\sigma^2 \gamma^2 \tilde{Q}^2}_{Z}.$$
(A2)

Again using (a) and (b) and taking the derivative of Equation (20), we get $A P'_\alpha B^T = B P'_\alpha A^T > 0$. Thus, $Z > 0$ and (A2) resembles a Lyapunov equation. Hence, we get $P''_\alpha > 0$. Thus, $J'_{priv}(\alpha, \sigma) = \frac{1}{2}tr(Q P'_\alpha) > 0$ and $J''_{priv}(\alpha, \sigma) = \frac{1}{2}tr(Q P''_\alpha) > 0$ and the proof is complete. ∎