

ON THE CONVERGENCE SPEED-ACCURACY TRADE-OFF IN COMPETING NETWORK LEARNING

Yike Zhao, Haoyuan Cai, Ali H. Sayed

École Polytechnique Fédérale de Lausanne, Switzerland

ABSTRACT

In this work, we study a stochastic two-team competition problem, where each team consists of a network of collaborating agents. Recent works have extended diffusion learning algorithms to this setting, including the adapt-then-combine and infer-then-combine (ATC-ITC) strategy, which employs a local inference step, and the competing diffusion (CD) strategy, which omits it. Although both methods have been shown to converge to an $\mathcal{O}(\mu)$ -neighborhood of the Nash equilibrium, existing analyses do not clearly characterize their relative advantages. To address this gap, we develop a unified framework for refined analysis of CD and ATC-ITC through a controlled inference step. The resulting performance bounds reveal a fundamental trade-off: the inference step in ATC-ITC accelerates convergence but introduces persistent gradient noise that can degrade steady-state performance relative to CD. This insight motivates a time-varying inference mechanism to balance the trade-off. Simulations validate the theoretical findings and the effectiveness of the proposed strategy.

Index Terms— Competing networks, diffusion learning, Nash equilibrium

1. INTRODUCTION

In many multi-agent systems, *cooperative* and *noncooperative* interactions coexist. Agents may collaborate within a team toward a common objective while simultaneously competing against other teams with conflicting interests. This setting gives rise to *competing networks problems*, where cooperation and competition are coupled across networks. For example, in Cournot-team competition, firms within each team coordinate their production strategies to maximize overall profit while competing against other teams through the quantities of goods supplied to the market [1]. Other applications include smart grid power management [2] and attack-defense scenarios in wireless networks [3]. More broadly, such problems can be modeled as two-network zero-sum games or multicluster games, where multiple networks interact competitively.

Many existing works on the competing networks problem focused on a *deterministic* setting, where exact gradients of the local risk functions are available. For two-network zero-sum games, the work [3] developed a distributed subgradient-based algorithm for time-varying graphs, while the work [4] introduced a distributed no-regret mirror descent method. The work [5] proposed an incremental strategy to relax the requirement of dense cross-team connections. However, it assumes that each agent knows the network Laplacian and incurs additional consensus steps. For multicluster games, the work [6] introduced a distributed gradient-tracking algorithm with linear convergence, but it restricts cross-team interactions to team leaders. The works [7, 8] considered directed graphs with identical and local step sizes, respectively, but rely on additional right-stochastic matrices for within-team cooperation, leading to extra computational steps [7].

Many real-world applications rely on noisy gradient oracles due to prohibitive computational costs, unknown data distributions, or environmental uncertainty. As a result, only *stochastic* gradients can be computed from random samples. In this setting, several diffusion learning algorithms have been proposed, including competing diffusion (CD), which extends the classical adapt-then-combine strategy to network games [9, 10], and adapt-then-combine and infer-then-combine (ATC-ITC), which improves the adversarial information estimation through an additional local inference step [1]. Although both algorithms are shown to converge to an $\mathcal{O}(\mu)$ neighborhood of the Nash equilibrium, their relative performance remains unclear due to coarse bounds derived under different analytical frameworks.

The contributions of this work are summarized as follows: 1) We develop a unified analytical framework and establish refined performance bounds for both CD and ATC-ITC under the same technical assumptions; 2) Our refined results reveal a fundamental trade-off: the additional inference step in ATC-ITC accelerates convergence but introduces extra gradient noise, leading to a larger steady-state error than CD, while CD converges more slowly but attains a lower steady-state error [9, 10]; 3) These unified and refined bounds motivate a time-varying inference strategy that balances convergence speed and steady-state accuracy.

Emails: yike.zhao@epfl.ch, haoyuan.cai@epfl.ch, ali.sayed@epfl.ch

2. PROBLEM FORMULATION

2.1. Network Formulation

We first review the network model used throughout the paper. Consider a collection of K agents divided into two teams of sizes K_1 and K_2 . Let $\mathcal{N}^{(1)} \triangleq \{1, \dots, K_1\}$ and $\mathcal{N}^{(2)} \triangleq \{K_1 + 1, \dots, K\}$ denote the index sets of Teams 1 and 2, respectively, where $K = K_1 + K_2$. Agents within each team cooperate through communication graphs induced by combination matrices $A^{(1)} \in \mathbb{R}^{K_1 \times K_1}$ and $A^{(2)} \in \mathbb{R}^{K_2 \times K_2}$, for Teams 1 and 2, respectively. For instance, the entry $a_{\ell k}^{(1)} > 0$ indicates a communication link from agent $\ell \in \mathcal{N}^{(1)}$ to agent $k \in \mathcal{N}^{(1)}$, and $a_{\ell k}^{(1)} = 0$ otherwise.

In addition to cooperative interactions within each team, agents also need to infer the adversary's strategy to make decisions through the inter-team combination matrix C :

$$C \triangleq \begin{bmatrix} C^{(1)} & C^{(12)} \\ C^{(21)} & C^{(2)} \end{bmatrix} \in \mathbb{R}^{K \times K}. \quad (1)$$

Here, the entry $c_{\ell k} > 0$ means that agent k receives information directly from agent ℓ , while $c_{\ell k} = 0$ means no direct link exists. The blocks $C^{(12)} \in \mathbb{R}^{K_1 \times K_2}$ and $C^{(21)} \in \mathbb{R}^{K_2 \times K_1}$ capture cross-team information flow, from Team 1 to Team 2 and from Team 2 to Team 1, respectively. The blocks $C^{(1)} \in \mathbb{R}^{K_1 \times K_1}$ and $C^{(2)} \in \mathbb{R}^{K_2 \times K_2}$ capture within-team communication, through which teammates share their inferred adversarial information.

To enable within-team cooperation and cross-team competition, we assume the combination matrices $A^{(1)}$, $A^{(2)}$ and C satisfy the following condition.

Assumption 1 (Connectivity). *We assume the combination matrices satisfy:*

- (a) *For $t \in \{1, 2\}$, the matrix $A^{(t)} \in \mathbb{R}^{K_t \times K_t}$ is primitive and left-stochastic, i.e. $\mathbb{1}^\top A^{(t)} = \mathbb{1}^\top$.*
- (b) *The matrix $C \in \mathbb{R}^{K \times K}$ is left-stochastic and irreducible (i.e., the underlying graph is strongly connected [11]).*

□

Note that Assumption 1b ensures that adversary information can flow across teams. It is weaker than the bipartite cross-network setup considered in [3, 12, 4] and is widely adopted in the competing networks literature [1, 9, 10, 2, 8, 7]. Assumption 1a is standard in distributed optimization to ensure consensus among agents [13]. This assumption further implies that $A^{(1)}$ and $A^{(2)}$ admit Perron eigenvectors $p^{(1)}$ and $p^{(2)}$ with positive entries satisfying $A^{(1)}p^{(1)} = p^{(1)}$, $A^{(2)}p^{(2)} = p^{(2)}$, and $\mathbb{1}_{K_1}^\top p^{(1)} = \mathbb{1}_{K_2}^\top p^{(2)} = 1$ [14]. We introduce

$$p \triangleq \begin{bmatrix} p^{(1)} \\ p^{(2)} \end{bmatrix}. \quad (2)$$

2.2. Two-Network Game Setting

We study a stochastic competing networks problem involving two networks with strategy variables $x \in \mathbb{R}^{M_1}$ and $y \in \mathbb{R}^{M_2}$, respectively. Each network aims to minimize its own global objective function defined as follows [1]:

$$\min_{x \in \mathbb{R}^{M_1}} J^{(1)}(x, y) \triangleq \sum_{k \in \mathcal{N}^{(1)}} p_k J_k(x, y), \quad (3a)$$

$$\min_{y \in \mathbb{R}^{M_2}} J^{(2)}(x, y) \triangleq \sum_{k \in \mathcal{N}^{(2)}} p_k J_k(x, y). \quad (3b)$$

Here, the weights $p_k > 0$ correspond to the entries of the vector p in (2) and satisfy $\sum_{k \in \mathcal{N}^{(t)}} p_k = 1$ for each team $t \in \{1, 2\}$. Therefore, $J^{(1)}$ and $J^{(2)}$ are weighted network-level objectives, while J_k is the local objective of agent k . This formulation includes two-network zero-sum games as a special case when $J^{(1)}(x, y) = -J^{(2)}(x, y)$. Moreover, we focus on a stochastic setting where the local objectives are of the form:

$$J_k(x, y) \triangleq \mathbb{E}\{Q_k(x, y; \xi_k)\}, k \in \mathcal{N}, \quad (4)$$

where $\mathcal{N} \triangleq \{\mathcal{N}^{(1)}, \mathcal{N}^{(2)}\}$ represents the index set of all agents. In this case, each agent k has access only to local samples ξ_k and can evaluate the loss value $Q_k(\cdot, \cdot; \xi_k)$.

For compactness, we define the following concatenated gradient vectors

$$F(z) \triangleq \begin{bmatrix} \nabla_x J^{(1)}(x, y) \\ \nabla_y J^{(2)}(x, y) \end{bmatrix}, \text{ where } z \triangleq \begin{bmatrix} x \\ y \end{bmatrix}, \quad (5)$$

where ∇_x (resp. ∇_y) denotes the gradient operator applied to argument x (resp. y). We assume the following condition holds for the gradient mapping operator:

Assumption 2 (Strong monotonicity). *The global gradient operator $F(z)$ is ν -strongly monotone, i.e., for every $z_1, z_2 \in \mathbb{R}^M$, where $M = M_1 + M_2$, we have*

$$(F(z_1) - F(z_2))^\top (z_1 - z_2) \geq \nu \|z_1 - z_2\|^2, \quad (6)$$

for a positive constant $\nu > 0$. □

Assumption 2 is widely adopted for studying the competing network problems [12, 1, 10, 6, 7, 8, 2] and generalizes the classical strongly convex–strongly concave min–max formulation. The goal of the network competing algorithms is to seek a Nash equilibrium defined as follows:

Definition 1. (Nash equilibrium [1, Definition 3]) *A point $z^* = [x^*; y^*]$ is a Nash equilibrium if*

$$J^{(1)}(x^*, y^*) \leq J^{(1)}(x, y^*), J^{(2)}(x^*, y^*) \leq J^{(2)}(x^*, y), \quad (7)$$

for all $x \in \mathbb{R}^{M_1}$ and $y \in \mathbb{R}^{M_2}$.

The following result from [15] guarantees the existence and uniqueness of the Nash equilibrium.

Lemma 1. ([15, Proposition 1.4.2, Theorem 2.3.3]) *Under Assumption 2, there exists a unique Nash equilibrium z^* for problem (3a)-(3b).*

3. A UNIFIED ANALYTICAL FRAMEWORK

In this section, we revisit two competing diffusion learning algorithms from a unified perspective. For clarity, we describe the update rule from the perspective of an agent $k \in \mathcal{N}^{(1)}$ in Team 1.

Within-team diffusion learning: At the start of each iteration i , agent $k \in \mathcal{N}^{(1)}$ uses its past strategy $\mathbf{x}_{k,i-1}$ and its local estimation of adversarial strategy $\mathbf{y}_{k,i-1}$ to perform the following update:

$$\phi_{k,i} = \mathbf{x}_{k,i-1} - \mu \widehat{\nabla_x J_k}(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1}), \quad (8)$$

$$\mathbf{x}_{k,i} = \sum_{\ell \in \mathcal{N}^{(1)}} a_{\ell k}^{(1)} \phi_{\ell,i}. \quad (9)$$

That is, the within-team learning process of an agent follows the standard adapt-then-combine (ATC) strategy from distributed optimization [14]. Here, μ is the step size, and $\widehat{\nabla_x J_k}$ denotes the stochastic local gradient of agent k , i.e.,

$$\widehat{\nabla_x J_k}(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1}) \triangleq \nabla_x Q_k(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1}; \boldsymbol{\xi}_{k,i}), \quad (10)$$

where $\boldsymbol{\xi}_{k,i}$ is the random sample used by agent k at iteration i .

Cross-team diffusion learning: Agent $k \in \mathcal{N}^{(1)}$ also updates its estimate of Team 2's strategy from its past information $\mathbf{y}_{k,i-1}$. In the following, we present the unified perspective by introducing a scaling factor $\kappa \geq 0$ to the key inference learning step, which uses the agent's local information to perform the inference update:

$$\psi_{k,i} = \mathbf{y}_{k,i-1} + \kappa \mu \widehat{\nabla_y J_k}(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1}), \quad (11)$$

where $\kappa \geq 0$ controls the strength of inference learning and is referred to as the *inference factor*. When $\kappa = 0$, the learning process reduces to that of the CD algorithm, whereas $\kappa = 1$ recovers the ATC-ITC algorithm. The rationale behind this inference step is that, under a zero-sum structure and assuming bounded gradient similarity [1], i.e.,

$$J^{(1)}(x, y) = -J^{(2)}(x, y), \quad (12)$$

$$\|\nabla_y J_k(x, y) - \nabla_y J^{(1)}(x, y)\| \leq c, \quad (13)$$

where c is a small constant, the local gradient knowledge $\nabla_y J_k(x, y)$ can serve as a proxy to $-\nabla_y J^{(2)}(x, y)$ in the sense that

$$\|\nabla_y J_k(x, y) - (-\nabla_y J^{(2)}(x, y))\| \leq c, \quad (14)$$

implying that the local gradient $\nabla_y J_k(x, y)$ can be used to estimate the adversarial information. However, in general games where $J^{(1)}(x, y) \neq -J^{(2)}(x, y)$, the mismatch $\|\nabla_y J_k(x, y) + \nabla_y J^{(2)}(x, y)\|$ can be large, making the inference step unreliable. In this case, setting $\kappa = 0$ disables inference updates, which in turn follows that of the CD algorithm. Incorporating the controlled inference learning step,

we obtain the cross-team learning step of both algorithms

$$\mathbf{y}_{k,i} = \sum_{\ell \in \mathcal{N}^{(2)}} c_{\ell k} \mathbf{y}_{\ell,i} + \sum_{\ell \in \mathcal{N}^{(1)}} c_{\ell k} \psi_{\ell,i}. \quad (15)$$

Note that the first term in (15) vanishes if agent k has no incoming links from nodes $\ell \in \mathcal{N}^{(2)}$, in which case $\mathbf{y}_{k,i}$ is entirely determined by inferred information from its neighbors. With this inference learning controlled by κ , we refer to the algorithm as adapt-then-combine and controlled infer-then-combine (ATC-CITC) algorithm, detailed in **Algorithm 1**.

For compactness, we summarize the algorithm by introducing the following network notation:

$$\mathcal{X}_i^1 \triangleq \text{col}\{\mathbf{x}_{1,i}, \dots, \mathbf{x}_{K_1,i}\} \in \mathbb{R}^{K_1 M_1 \times 1}, \quad (16)$$

$$\mathcal{Y}_i^1 \triangleq \text{col}\{\mathbf{y}_{1,i}, \dots, \mathbf{y}_{K_1,i}\} \in \mathbb{R}^{K_1 M_2 \times 1}, \quad (17)$$

and $\mathcal{Y}_i^2 \in \mathbb{R}^{K_2 M_2 \times 1}$, $\mathcal{X}_i^2 \in \mathbb{R}^{K_2 M_1 \times 1}$ are defined similarly for Team 2. The superscripts are used to distinguish the network quantities associated with Teams 1 and 2. For instance, $\mathcal{Y}_i^1$ and $\mathcal{X}_i^2$ represent inferred adversarial information by Teams 1 and 2, respectively. In addition, we define the networked stochastic gradients:

$$\mathcal{G}_{x,i}^1 \triangleq \text{col}\{\widehat{\nabla_x J_k}(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1})\}_{k=1}^{K_1}, \quad (18)$$

$$\mathcal{G}_{y,i}^1 \triangleq \text{col}\{-\widehat{\nabla_y J_k}(\mathbf{x}_{k,i-1}, \mathbf{y}_{k,i-1})\}_{k=1}^{K_1}. \quad (19)$$

The quantities $\mathcal{G}_{x,i}^2$ and $\mathcal{G}_{y,i}^2$ are defined in the same way. The augmented combination matrices are defined as follows:

$$\begin{aligned} \mathcal{C}^{(1)} &\triangleq C^{(1)} \otimes I_{M_2}, & \mathcal{C}^{(2)} &\triangleq C^{(2)} \otimes I_{M_1}, \\ \mathcal{C}^{(21)} &\triangleq C^{(21)} \otimes I_{M_2}, & \mathcal{C}^{(12)} &\triangleq C^{(12)} \otimes I_{M_1}, \\ \mathcal{A}^{(1)} &\triangleq A^{(1)} \otimes I_{M_1}, & \mathcal{A}^{(2)} &\triangleq A^{(2)} \otimes I_{M_2}. \end{aligned} \quad (20)$$

The final algorithm is summarized in **Algorithm 1**.

Algorithm 1 Adapt-then-Combine and Controlled Infer-then-Combine (ATC-CITC) algorithm

- 1: **Initialize:** strategies $\mathcal{X}_{-1}^1, \mathcal{Y}_{-1}^1, \mathcal{X}_{-1}^2, \mathcal{Y}_{-1}^2 \leftarrow 0$, adaptation step size $\mu > 0$, inference factor $\kappa \geq 0$
 - 2: **for** $i = 0, \dots$ **do**
 - 3: Within-team diffusion
 $\mathcal{X}_i^1 = \mathcal{A}^{(1)\top} (\mathcal{X}_{i-1}^1 - \mu \mathcal{G}_{x,i}^1)$
 $\mathcal{Y}_i^2 = \mathcal{A}^{(2)\top} (\mathcal{Y}_{i-1}^2 - \mu \mathcal{G}_{y,i}^2)$
 - 4: Cross-team diffusion
 $\psi_i^1 = \mathcal{Y}_{i-1}^1 - \kappa \mu \mathcal{G}_{y,i}^1$
 $\psi_i^2 = \mathcal{X}_{i-1}^2 - \kappa \mu \mathcal{G}_{x,i}^2$
 $\mathcal{Y}_i^1 = \mathcal{C}^{(21)\top} \mathcal{Y}_i^2 + \mathcal{C}^{(1)\top} \psi_i^1$
 $\mathcal{X}_i^2 = \mathcal{C}^{(12)\top} \mathcal{X}_i^1 + \mathcal{C}^{(2)\top} \psi_i^2$
 - 5: **end for**
-

4. CONVERGENCE ANALYSIS

4.1. Assumptions

We introduce some standard assumptions to facilitate the theoretical analysis.

Assumption 3 (Lipschitz gradients). For each $t \in \{1, 2\}$ and $k \in \mathcal{N}^{(t)}$, we assume the gradients associated with each local risk function $J_k(\cdot, \cdot)$ are L_f -Lipschitz, i.e., $\forall x_1, x_2 \in \mathbb{R}^{M_1}, y_1, y_2 \in \mathbb{R}^{M_2}$:

$$\begin{aligned} \|\nabla_w J_k(x_1, y_1) - \nabla_w J_k(x_2, y_2)\| \\ \leq L_f (\|x_1 - x_2\| + \|y_1 - y_2\|), \end{aligned} \quad (21)$$

where $w = x$ or y . $\square$

Assumption 4 (Bounded gradient disagreement). For each $t \in \{1, 2\}$ and $k \in \mathcal{N}^{(t)}$, the gradient disagreement between the local risk functions and the global risk function is bounded, i.e., $\forall x_1 \in \mathbb{R}^{M_1}, y_1 \in \mathbb{R}^{M_2}$:

$$\|\nabla_w J_k(x_1, y_1) - \nabla_w J^{(t)}(x_1, y_1)\| \leq G, \quad (22)$$

where $w = x$ or y . $\square$

Assumption 5 (Gradient noise process). We define the filtration generated by the random processes as $\mathcal{F}_i = \{(\mathbf{x}_{k,j}, \mathbf{y}_{k,j}) \mid k = 1, \dots, K, j = -1, \dots, i\}$. For each $t \in \{1, 2\}$ and $k \in \mathcal{N}^{(t)}$, we assume the stochastic gradients are unbiased with bounded variance conditioned on $\mathcal{F}_i$, i.e., $\forall \mathbf{x}, \mathbf{y} \in \mathcal{F}_i$,

$$\mathbb{E}\{\widehat{\nabla_w J_k(\mathbf{x}, \mathbf{y})} \mid \mathcal{F}_i\} = \nabla_w J_k(\mathbf{x}, \mathbf{y}), \quad (23)$$

$$\mathbb{E}\{\|\widehat{\nabla_w J_k(\mathbf{x}, \mathbf{y})} - \nabla_w J_k(\mathbf{x}, \mathbf{y})\|^2 \mid \mathcal{F}_i\} \leq \sigma^2, \quad (24)$$

where $w = x$ or y . $\square$

Note that similar assumptions on gradient disagreement and gradient noise are also adopted in distributed stochastic optimization works such as [12, 1, 16, 17].

4.2. Main Results

To facilitate analysis, we introduce the following notation:

$$\begin{aligned} \mathbf{x}_{c,i} &\triangleq \sum_{k \in \mathcal{N}^{(1)}} p_k \mathbf{x}_{k,i}, & \mathbf{y}_{c,i} &\triangleq \sum_{k \in \mathcal{N}^{(2)}} p_k \mathbf{y}_{k,i}, \\ \mathcal{X}_{c,i} &\triangleq \mathbb{1}_{K_1} \otimes \mathbf{x}_{c,i}, & \mathcal{Y}_{c,i} &\triangleq \mathbb{1}_{K_2} \otimes \mathbf{y}_{c,i}, \\ \mathcal{X}'_{c,i} &\triangleq \mathbb{1}_{K_2} \otimes \mathbf{x}_{c,i}, & \mathcal{Y}'_{c,i} &\triangleq \mathbb{1}_{K_1} \otimes \mathbf{y}_{c,i}. \end{aligned} \quad (25)$$

The variables $\mathbf{x}_{c,i}$ and $\mathbf{y}_{c,i}$ represent the centroid strategies of Teams 1 and 2, obtained as weighted averages of the strategies within each team. Since the local agents can cluster around the centroid under a small learning rate, we use $\mathbf{x}_{c,i}$ and $\mathbf{y}_{c,i}$ as a “proxy” for studying the learning dynamics of the algorithm. The variable $\mathcal{X}_{c,i} \in \mathbb{R}^{K_1 M_1}$ is constructed by repeating $\mathbf{x}_{c,i}$ for K_1 times, while $\mathcal{X}'_{c,i} \in \mathbb{R}^{K_2 M_1}$ is constructed

by repeating $\mathbf{x}_{c,i}$ for K_2 times. The quantities $\mathcal{Y}_{c,i}, \mathcal{Y}'_{c,i}$ are constructed in a similar way. We can first establish a result similar to [1].

Lemma 2. ([1, Lemma 5]) Under Assumption 1, the consensus error satisfies

$$\|\mathcal{X}_i^1 - \mathcal{X}_{c,i}\|^2 + \|\mathcal{X}_i^2 - \mathcal{X}'_{c,i}\|^2 \leq \|\mathcal{P}_L\|^2 \left\| \mathcal{P}_R^\top \begin{bmatrix} \mathcal{X}_i^1 \\ \mathcal{X}_i^2 \end{bmatrix} \right\|^2, \quad (26)$$

$$\|\mathcal{Y}_i^1 - \mathcal{Y}'_{c,i}\|^2 + \|\mathcal{Y}_i^2 - \mathcal{Y}_{c,i}\|^2 \leq \|\mathcal{Q}_L\|^2 \left\| \mathcal{Q}_R^\top \begin{bmatrix} \mathcal{Y}_i^1 \\ \mathcal{Y}_i^2 \end{bmatrix} \right\|^2, \quad (27)$$

where the matrices $\mathcal{P}_L, \mathcal{P}_R, \mathcal{Q}_L$, and $\mathcal{Q}_R$ are obtained from the spectral decomposition of $A^{(1)}, A^{(2)}$, and C . Their explicit expressions are given in (54)–(55) of [1].

We then define a coupled error that encodes mean-square and consensus error as follows:

$$\Delta_i \triangleq \mathbb{E} \left\{ \underbrace{\|\mathbf{z}_{c,i} - \mathbf{z}^*\|^2}_{\text{mean-square error}} + \underbrace{\left\| \mathcal{P}_R^\top \begin{bmatrix} \mathcal{X}_i^1 \\ \mathcal{X}_i^2 \end{bmatrix} \right\|^2 + \left\| \mathcal{Q}_R^\top \begin{bmatrix} \mathcal{Y}_i^1 \\ \mathcal{Y}_i^2 \end{bmatrix} \right\|^2}_{\text{scaled consensus error}} \right\}. \quad (28)$$

Here, the first term measures the squared error of the centroid $\mathbf{z}_{c,i} = [\mathbf{x}_{c,i}; \mathbf{y}_{c,i}]$ from the solution point, while the second bounds represent the scaled consensus error.

Theorem 1. Let Assumptions 1, 2, 3, 4, 5 hold for ATC-CITC and further assume the objectives follow a zero-sum structure (12). We choose step size μ sufficiently small:

$$\mu < \mu_o, \quad (29)$$

where μ_o depends on κ, ν, L_f , and network-related quantities. Then, the coupled error term Δ_i decays at the following rate:

$$\Delta_i \leq d \Delta_{i-1} + \mathcal{O}((1 + \kappa^2)\mu^2), \quad (30)$$

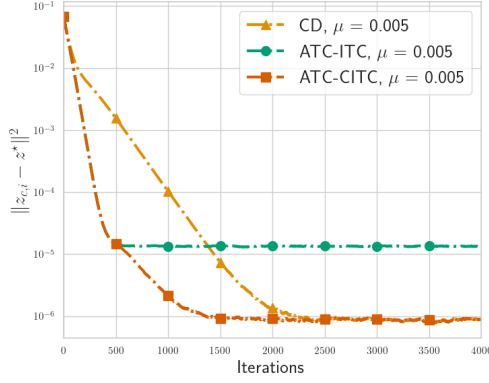
where d is a constant given by

$$\begin{aligned} d = & (1 + \nu\mu) (1 - 2\mu\nu + 4L_f^2\mu^2) \\ & + \underbrace{8(c_1 + c_2)(1 - \kappa)^2\mu^2}_{\text{positive if } \kappa \neq 1} < 1. \end{aligned} \quad (31)$$

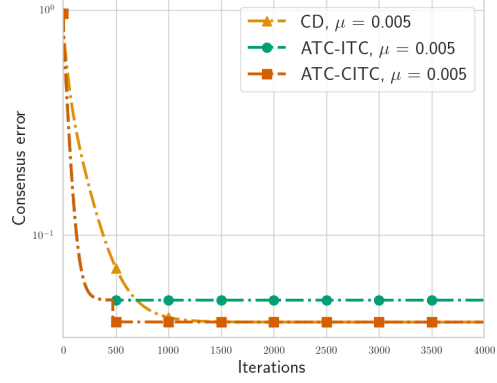
Here, c_1 and c_2 depend on the network structures. In particular, when $\kappa \neq 1$, the last term of d is nonzero, leading to slower convergence compared to the case $\kappa = 1$. Furthermore, it holds asymptotically that

$$\begin{aligned} \limsup_{i \rightarrow \infty} \Delta_i \\ \leq \mathcal{O} \left(\mu K (\sigma^2 + G^2) + \underbrace{\kappa^2 \mu K (\sigma^2 + G^2)}_{\text{positive if } \kappa \neq 0} \right). \end{aligned} \quad (32)$$

When $\kappa \neq 0$, the last term acts as a persistent error, resulting in a larger overall error compared to the case $\kappa = 0$.



(a) Mean-square error.



(b) Consensus error.

Fig. 1: Comparison of algorithms for a synthesized quadratic zero-sum game.

Proof. The proof is omitted for brevity. $\square$

Theorem 1 establishes the convergence of **ATC-CITC** for any $\kappa \geq 0$ and reveals the role of the inference learning, with **ATC-ITC** corresponding to $\kappa = 1$ and **CD** to $\kappa = 0$. The result shows a fundamental trade-off: inference learning in **ATC-ITC** yields a smaller convergence factor d , leading to a faster convergence than **CD**; however, it also introduces an additional noisy error at each iteration (see (30)). This extra error degrades the steady-state performance.

We then discuss the previously unstudied regime. For $0 < \kappa < 1$, a suitable choice of κ can balance the performance of **ATC-ITC** and **CD**. In contrast, choosing $\kappa > 1$ leads to slower convergence and a larger steady-state error than **ATC-ITC**. We also find that this setting narrows the choice of the step size μ , indicating a detrimental effect on the choice of μ when $\kappa > 1$. Although Theorem 1 focuses on the zero-sum case, it can readily be extended to *general-sum objectives*, i.e., $J^{(1)} \neq J^{(2)}$, by setting $\kappa = 0$, where active inference may no longer be reliable.

Corollary 1. *Let Assumptions 1, 2, 3, 4, 5 hold for **CD** (**ATC-CITC** with $\kappa = 0$). Suppose the step size satisfies $\mu < \mu_o$ under $\kappa = 0$. Then, the centroid $z_{c,i}$ asymptotically converges to the Nash equilibrium z^* in the mean-square-error sense*

$$\limsup_{i \rightarrow \infty} \Delta_i \leq \mathcal{O}(\mu K (\sigma^2 + G^2)). \quad (33)$$

Proof. The proof follows by setting $\kappa = 0$, without the need for the zero-sum assumption. $\square$

This result shows that **CD** converges to an $\mathcal{O}(\mu)$ neighborhood of the Nash equilibrium under a general game setting, consistent with existing works [9, 10].

Knowing how κ affects convergence speed and steady-state accuracy, a natural question is how to balance convergence speed and steady-state error. Motivated by the theory, we adopt time-varying inference factors: using a larger value

in the early stage to accelerate convergence and gradually reducing it to improve steady-state performance. We present a simple strategy below, while noting that other schedules are also possible. Let κ_i denote the iteration-dependent factor.

Theorem 2. *Let Assumptions 1, 2, 3, 4, 5 hold for **ATC-CITC** under zero-sum objectives (12). Suppose the sequence $\{\kappa_i\}_{i \geq 0}$ is non-increasing with*

$$\kappa_0 = 1, \quad \lim_{i \rightarrow \infty} \kappa_i = 0, \quad (34)$$

and the step size satisfies $\mu < \mu_o$ under both cases $\kappa = 0$ and $\kappa = 1$. Then, the coupled error term Δ_i decays at the following form:

$$\Delta_i \leq d_i \Delta_{i-1} + \mathcal{O}((1 + \kappa_i^2)\mu^2), \quad (35)$$

where d_i is given by

$$d_i = (1 + \nu\mu) (1 - 2\mu\nu + 4L_f^2\mu^2) + 8(c_1 + c_2)(1 - \kappa_i)^2\mu^2 < 1. \quad (36)$$

Furthermore, it holds asymptotically that

$$\limsup_{i \rightarrow \infty} \Delta_i \leq \mathcal{O}(\mu K (\sigma^2 + G^2)). \quad (37)$$

$\square$

We can see that setting $\kappa_i = 1$ yields a convergence rate comparable to **ATC-ITC** in the transient-time regime. As the iterations evolve, κ_i approaches zero, deactivating the inference learning and leading to improved steady-state performance comparable to that of **CD**; see (33) and (37).

5. NUMERICAL EXPERIMENT

We consider a synthesized quadratic zero-sum game similar to [12]. Let $x, y \in \mathbb{R}^{20}$ and $K_1 = K_2 = 6$. The local risks for Team 1 are defined as

$$J_k(x, y) \triangleq \mathbb{E}_{\xi_k} \{x^\top y + \|x - a_k\|^2 - \|y - b_k\|^2 + x^\top g_x(\xi_k) + g_y^\top(\xi_k)y\}, \quad k \in \mathcal{N}^{(1)}. \quad (38)$$

Here, $g_x(\xi_k)$ is a Gaussian random vector drawn from $\mathcal{N}(0, \sigma^2 I)$, where $\sigma = 10^{-2}$. The vector $g_y(\xi_k)$ is generated from the same distribution. For each agent $k \in \mathcal{N}^{(1)}$, there is a corresponding agent $k' \in \mathcal{N}^{(2)}$ whose local risk satisfies $J_{k'}(x, y) = -J_k(x, y)$. All random variables are spatially and temporally independent. Two doubly stochastic matrices are generated for $A^{(1)}$ and $A^{(2)}$ using the Metropolis-Hastings rule, yielding uniform vectors $p^{(1)}$ and $p^{(2)}$. A connected graph is used for C . This game satisfies Assumptions 1–5. For **ATC-CITC**, we consider the stagewise rule

$$\kappa_i = \begin{cases} 1, & i \leq S \quad (= 470) \\ 0, & \text{otherwise} \end{cases}, \quad (39)$$

which employs inference learning during the first S iterations and drops it thereafter. The code is available online¹.

The simulation results are plotted in Fig. 1, which illustrates a clear trade-off between convergence speed and steady performance for $\kappa = 1$ (**ATC-ITC**) and $\kappa = 0$ (**CD**). With the proposed stagewise inference factor, **ATC-CITC** matches the convergence speed of **ATC-ITC** and steady-state error of **CD**, achieving superior performance.

6. CONCLUSION

In this paper, we developed a unified framework to analyze two diffusion competing learning methods for stochastic competing network problems, namely, **ATC-ITC** and **CD**. We established a unified and refined performance bound for both algorithms. Our theoretical results reveal an intrinsic trade-off on the role of inference learning step: the inference learning in **ATC-ITC** accelerates convergence but introduces persistent gradient noise, which can degrade its steady-state performance relative to **CD**. We further develop a time-varying inference strategy to balance this trade-off. Simulations on a synthesized example validated our theoretical findings.

7. REFERENCES

- [1] Y. Zhao, H. Cai, and A. H. Sayed, “Diffusion stochastic learning over adaptive competing networks,” *IEEE Transactions on Signal Processing*, vol. 74, pp. 1141–1155, 2026.
- [2] R. Yu, M. Meng, and L. Li, “Distributed online learning for leaderless multicluster games in dynamic environments,” *IEEE Transactions on Control of Network Systems*, vol. 11, no. 3, pp. 1548–1561, 2023.
- [3] Y. Lou, Y. Hong, L. Xie, G. Shi, and K. H. Johansson, “Nash equilibrium computation in subnetwork zero-sum games with switching communications,” *IEEE Transactions on Automatic Control*, vol. 61, no. 10, pp. 2920–2935, 2015.
- [4] S. Huang, J. Lei, Y. Hong, U. V. Shanbhag, and J. Chen, “No-regret distributed learning in subnetwork zero-sum games,” *IEEE Transactions on Automatic Control*, vol. 69, no. 10, pp. 6620–6635, 2024.
- [5] C. X. Shi and G. H. Yang, “Nash equilibrium computation in two-network zero-sum games: An incremental algorithm,” *Neurocomputing*, vol. 359, pp. 114–121, 2019.
- [6] M. Meng and X. Li, “On the linear convergence of distributed Nash equilibrium seeking for multi-cluster games under partial-decision information,” *Automatica*, vol. 151, pp. 110919, 2023.
- [7] J. Zimmermann, T. Tatarenko, V. Willert, and J. Adamy, “Solving leaderless multi-cluster games over directed graphs,” *European Journal of Control*, vol. 62, pp. 14–21, 2021.
- [8] Y. Pang and G. Hu, “Distributed Nash equilibrium seeking in n-cluster games with non-uniform constant step-sizes,” in *Proc. IEEE Conference on Decision and Control (CDC)*, 2023, pp. 4182–4188.
- [9] S. Vlaski and A. H. Sayed, “Competing adaptive networks,” in *Proc. IEEE Statistical Signal Processing Workshop (SSP)*, 2021, pp. 71–75.
- [10] V. Shashkov, F. R. M. Pavan, H. Cai, and A. H. Sayed, “Stability and performance analysis of diffusion learning for two-network competing problems,” in *Proc. IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2025, pp. 1–6.
- [11] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*, SIAM, 2023.
- [12] Y. Zhao, H. Cai, and A. H. Sayed, “Diffusion learning over adaptive competing networks,” in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2025, pp. 1–5.
- [13] A. H. Sayed, *Inference and Learning from Data*, Cambridge University Press, 2022.
- [14] A. H. Sayed, “Adaptation, learning, and optimization over networks,” *Foundations and Trends® in Machine Learning*, vol. 7, no. 4-5, pp. 311–801, 2014.
- [15] F. Facchinei, *Finite-Dimensional Variational Inequalities and Complementarity Problems*, Springer, 2003.
- [16] S. Vlaski and A. H. Sayed, “Distributed learning in non-convex environments—part i: Agreement at a linear rate,” *IEEE Transactions on Signal Processing*, vol. 69, pp. 1242–1256, 2021.
- [17] H. Cai, S. A. Alghunaim, and A. H. Sayed, “Diffusion stochastic optimization for min-max problems,” *IEEE Transactions on Signal Processing*, vol. 73, pp. 259–274, 2025.

¹<https://github.com/yikezhao0812/Competing-Networks>