# ADAPTIVE DIFFUSION WITH COMPRESSED COMMUNICATION

*Marco Carpentiero*$^{\star}$ *Vincenzo Matta*$^{\star}$ *Ali H. Sayed*$^{\dagger}$

$^{\star}$ DIEM, University of Salerno, Fisciano (SA), Italy
$^{\dagger}$ EPFL, School of Engineering, CH-1015 Lausanne, Switzerland

## ABSTRACT

We consider multi-agent networks that aim at solving, cooperatively and online, distributed optimization problems under communication constraints. We propose the ACTC (Adapt-Compress-Then-Combine) diffusion strategy, which leverages *differential randomized compression* to infuse the classical ATC strategy with the ability to handle compressed data. We consider the flexible setting of directed graphs and left-stochastic policies, and require strong convexity only at a network level (i.e., some agents might even have non-convex risks). We prove that each agent is able to learn the optimal solution up to a small error on the order of the step-size, achieving remarkable savings in terms of bits exchanged between neighboring agents.

***Index Terms***— Distributed optimization, adaptation and learning, randomized quantizers, differential quantization.

## 1. INTRODUCTION AND RELATED WORK

The continuous advances in the fields of statistical learning and network science have made *distributed optimization over networks* a steadily growing research area [1–10]. Distributed strategies offer several advantages, such as scalability, possibility of working with reduced-size and spatially dispersed datasets, and robustness to failures. In addition, distributed cooperation allows the agents to overcome their individual limitations, and to deliver superior performance w.r.t. single-agent strategies [11, 12]. Since cooperation requires exchange of information across spatially separated agents, and since the transmitted information usually needs to be compressed to meet communication constraints, *data compression* becomes a critical part in the design of distributed strategies.

One critical difficulty in data compression for inferential problems is the lack of knowledge about the data distributions, since the latter depend on the same *unknown* parameters that the agents are trying to learn. This fact complicates the quantizer design. A breakthrough in this problem is *randomized quantization*, whose foundations can be traced back to the seminal works [13, 14], while its potential for distributed inference was exploited in [15–17]. More recently, the idea was applied to distributed optimization by means of stochastic gradient algorithms in [18].

However, quantization errors can accumulate over successive iterations and impair the convergence of distributed strategies. By calling upon the theory of differential/predictive quantization (e.g., the $\Sigma\Delta$ modulation scheme adopted in PCM [19]), the impact of quantization errors can be reduced by leveraging the memory of recursive implementations such as gradient descent [20–23]. *Differential quantization* aims at reducing the error variance by compressing only the difference (i.e., the innovation) between subsequent iterates. This is advantageous under a fixed budget of quantization bits since the innovation typically exhibits a reduced range as compared to the entire sample. Moreover, in view of the correlation that exists between consecutive samples, quantizing the entire sample would waste resources by transmitting redundant information. Some information-theoretic bounds on the performance of *non-stochastic* gradient descent under differential quantization appear in [24].

All the aforementioned works dealing with data compression for distributed optimization focus on the case where agents communicate with a fusion center, which carries out centralized gradient update steps. In this work, we focus instead on *adaptive networks*, namely, on *fully decentralized* architectures where each agent $i)$ is responsible for its own inference obtained through local interactions with its neighbors; and $ii)$ collects noisy streaming data to evaluate *stochastic instantaneous approximations* of the actual gradients, and must respond in real time to drifts in the underlying conditions. *Without* compression constraints, typical adaptive strategies are consensus or diffusion implementations [11]. Extending these strategies to handle compressed data is nontrivial, since without proper design of the quantizers, significant bias would be introduced into the learning algorithms. One early characterization of adaptive diffusion under communication constraints was provided in [25], where the errors arising from compression are modeled as noise over the communication links.

Recent studies took into account the explicit encoder structure. In particular, consensus strategies based on differential randomized quantization were proposed in [26], albeit with focus on the *diminishing* step-size regime, which is not suited to the adaptive setting [11]. Results about the regime of *constant* step-sizes are available in [27], with reference to a primal-dual consensus strategy, undirected graphs with symmetric and doubly-stochastic combination policies, and strongly-convex cost functions for all agents.

We can now list the main contributions offered in this work. Our results (summarized in Theorem 1 further ahead) allow a thorough characterization of the learning dynamics of *adaptive diffusion* over communication-constrained networks. The analysis is carried out under flexible assumptions, such as *directed graphs* and *left-stochastic* combination policies, and a strong convexity assumptions requested only at a *network level*, namely, the cost functions of the individual agents need not be convex, provided that a suitable global cost function is strongly convex.

**Notation**. Boldface letters denote random variables and normal font letters their realizations. All vectors are column vectors. The symbol $\mathbb{1}_L$ is the $L \times 1$ vector with entries equal to 1, and $I_L$ is the identity matrix of size $L$. For two square matrices $X$ and $Y$, the notation $X \geq Y$ signifies that $X - Y$ is positive semidefinite. The symbol $\mathbb{E}$ denotes the expectation operator. For a nonnegative function $f(\mu)$, the notation $f(\mu) = O(\mu)$ signifies that $\exists C > 0, \mu_0 > 0$ such that $f(\mu) \leq C\mu$ for all $\mu \leq \mu_0$.

## 2. MODEL AND ASSUMPTIONS

We consider a network of $N$ agents solving a distributed optimization problem. Each agent $k = 1, 2, \dots, N$, is assigned a *local* cost function $J_k(w) : \mathbb{R}^M \to \mathbb{R}$, satisfying the following condition.

**Assumption 1** (Smoothness)**.** *For all $w \in \mathbb{R}^M$, each cost function $J_k(w)$ is twice-differentiable and its Hessian matrix satisfies the Lipschitz condition $\nabla^2 J_k(w) \le \eta_k I_M$, for some constant $\eta_k > 0$.* □

One popular distributed optimization mechanism is the Adapt-Then-Combine (ATC) diffusion strategy [4, 11]:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu_k \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) & \text{[Adapt]} \\ \boldsymbol{w}_{k,i} = \displaystyle\sum_{\ell=1}^{N} a_{\ell k} \boldsymbol{\psi}_{\ell,i} & \text{[Combine]} \end{cases} \tag{1}$$

In (1), agents $k = 1, 2, \dots, N$, evolve over time $i = 1, 2, \dots$, by producing a sequence of iterates $\boldsymbol{w}_{k,i} \in \mathbb{R}^M$. The adaptation step is a *self-learning* step, where each agent $k$ at time $i$ computes a *stochastic instantaneous approximation* $\widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$ of the (unavailable) true gradient. Such approximation is scaled by a small step-size $\mu_k > 0$ to update $\boldsymbol{w}_{k,i-1}$ following the (stochastic) gradient descent. The maximum step-size and the *scaled* step-sizes will be denoted by, respectively:

$$\mu \triangleq \max_{k=1,2,\dots,N} \mu_k, \qquad \alpha_k \triangleq \mu_k/\mu. \tag{2}$$

The combination step is a *social learning* step, where agent $k$ combines the updated states received from its neighbors. In particular, we model the links between agents through a *directed* graph. The graph edges (i.e., links) are characterized by weights collected into the combination matrix $A = [a_{\ell k}]$. The combination process is a local process where only neighboring agents interact, with the neighborhood of agent $k$ being $\mathcal{N}_k = \{\ell = 1, 2, \dots, N : a_{\ell k} > 0\}$.

**Assumption 2** (Strongly-Connected Network)**.** *Given any pair of nodes $(\ell, k)$, paths with nonzero weights exist in both directions (i.e., from $\ell$ to $k$ and vice versa), and at least one agent $k$ in the entire network has a self-loop ($a_{kk} > 0$).* □

**Assumption 3** (Left-Stochastic Policy)**.** *The entries of the combination matrix $A$ fulfill the following conditions, for $k = 1, 2, \dots, N$:*

$$a_{\ell k} \ge 0, \quad \sum_{\ell \in \mathcal{N}_k}^{N} a_{\ell k} = 1, \quad a_{\ell k} = 0 \quad \text{for } \ell \notin \mathcal{N}_k. \tag{3}$$

□

Under Assumptions 2 and 3, matrix $A$ is primitive. In view of the Perron-Frobenius theorem, this implies the existence of the *Perron eigenvector* $\pi = [\pi_1, \pi_2, \dots, \pi_N]^\top$, having strictly positive entries and satisfying the equalities $A\pi = \pi$ and $\mathbb{1}_N^\top \pi = 1$.

The ATC strategy was characterized in great detail in previous works [4–6, 11]. In particular, it was shown that, for sufficiently small $\mu$, the ATC iterates of all agents converge to a small neighborhood of the value $w^\star$ that minimizes the following *global cost function*:

$$J(w) = \sum_{k=1}^{N} p_k J_k(w), \tag{4}$$

where $p_k = \alpha_k \pi_k$. With doubly-stochastic policies and equal step-sizes (i.e., with $p_k = 1/N$ for all $k$), minimizing $J(w)$ amounts to minimizing the plain sum of the local cost functions. On the other hand, left-stochastic policies and non-identical step-sizes open several additional possibilities as regards to tuning the weights $\{p_k\}$, which can be useful, e.g., to explore different Pareto solutions to suitable multi-objective problems [28]. Remarkably, the ATC analysis carried out in [5,6,11] does not assume convexity of all local cost functions $J_k(w)$, relying only on the following *global* condition.

**Assumption 4** (Global Strong Convexity)**.** *The aggregate cost function $J(w)$ in* (4) *is $\nu$-strongly convex, namely, a positive constant $\nu$ exists such that $\sum_{k=1}^{N} p_k \nabla^2 J_k(w) \ge \nu\, I_M$.* □

For example, global strong convexity can be satisfied even when only a single agent has a strongly-convex cost function, with the other agents having possibly non-convex functions.

## 3. ACTC DIFFUSION STRATEGY

The ATC strategy assumes that perfectly reliable information is exchanged between agents (i.e., no compression). We now introduce the ACTC diffusion strategy, which incorporates data compression. The ACTC time-evolution can be described through three time-varying variables: an intermediate update $\boldsymbol{\psi}_{k,i}$, a differentially-quantized update $\boldsymbol{q}_{k,i}$, and the current minimizer $\boldsymbol{w}_{k,i}$. At time $i = 0$, each agent $k$ is initialized with an arbitrary state $\boldsymbol{q}_{k,0}$. Then, agent $k$ receives $\{\boldsymbol{q}_{\ell,0}\}$ from its neighbors $\ell \in \mathcal{N}_k$ (such initial sharing is performed with infinite precision, which is immaterial to our analysis since it happens only once), and computes an initial minimizer $\boldsymbol{w}_{k,0} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{q}_{\ell,0}$. Then, for every $i > 0$, each agent $k$ performs the operations:

$$\begin{cases} \boldsymbol{\psi}_{k,i} = \boldsymbol{w}_{k,i-1} - \mu_k \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1}) \\ \boldsymbol{q}_{\ell,i} = \boldsymbol{q}_{\ell,i-1} + \zeta\, \boldsymbol{Q}_\ell(\boldsymbol{\psi}_{\ell,i} - \boldsymbol{q}_{\ell,i-1}) \quad \forall \ell \in \mathcal{N}_k \\ \boldsymbol{w}_{k,i} = \displaystyle\sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{q}_{\ell,i} \end{cases} \tag{5}$$

Comparing (5) against (1), we see the appearance of an intermediate *compression* step, where agent $k$ receives from its neighbors $\ell \in \mathcal{N}_k$ the *compressed* values $\boldsymbol{Q}_\ell(\boldsymbol{\psi}_{\ell,i} - \boldsymbol{q}_{\ell,i-1})$. The bold font used for $\boldsymbol{Q}_\ell(\cdot)$ highlights that *random* compression operators are permitted. The compression step shows clearly the usage of *differential quantization*, since we see that the operator $\boldsymbol{Q}_\ell(\cdot)$ encodes only the innovation, namely, the difference between the update $\boldsymbol{\psi}_{\ell,i}$ and the old state $\boldsymbol{q}_{\ell,i-1}$. Similar forms of error compensation are not unique to schemes with *compressed* data. One useful example is *exact diffusion* [29, 30] (without compression), where the true gradient is exactly available, and compensation is used to let the mean-square deviation vanish even with constant step-size. Since the quantization operation is applied to *differences*, the states $\boldsymbol{q}_{\ell,i}$ must be updated by adding the compressed difference to the previous value[1] $\boldsymbol{q}_{\ell,i-1}$. Such update is performed through a weighting parameter $\zeta \in (0, 1)$, which is useful to control the stability of the algorithm. Finally, agent $k$ *combines* linearly the updated quantized states received from its neighbors, scaled by the convex combination weights $a_{\ell k}$.

### 3.1. Compression operators

**Assumption 5** (Randomized Compression Operators)**.** *Given an input value $x \in \mathbb{R}^M$, the randomized operator $\boldsymbol{Q} : \mathbb{R}^M \to \mathbb{R}^M$*

[1] For all $\ell \in \mathcal{N}_k$, the value $\boldsymbol{q}_{\ell,i-1}$ is known to agent $k$, since the initial value $\boldsymbol{q}_{\ell,0}$ is known, and the subsequent values up to $\boldsymbol{q}_{\ell,i-1}$ can be iteratively constructed by storing only the most recent quantized difference.

*satisfies the following properties, for a certain $\omega > 0$:*

$$\mathbb{E}\left[\boldsymbol{Q}(x)\right] = x \qquad \text{[unbiasedness]} \tag{6}$$

$$\mathbb{E}\,\|\boldsymbol{Q}(x) - x\|^2 \leq \omega\,\|x\|^2 \quad \text{[non blow-up property]} \tag{7}$$

□

When we say that the operator is randomized we mean that, for a deterministic input $x$, the output $\boldsymbol{Q}(x)$ is a random variable. The expectations in (6)-(7) are accordingly evaluated w.r.t. the randomness inherent to $\boldsymbol{Q}(\cdot)$. Whenever we apply the operator to a random input $\boldsymbol{x}$, we assume that the random mechanism governing $\boldsymbol{Q}(\cdot)$ is independent of $\boldsymbol{x}$. Since we allow each agent $k$ to employ a different compression operator $\boldsymbol{Q}_k(x)$, we can have different compression parameters $\omega_k$, with their maximum value being denoted by:

$$\Omega \triangleq \max_{k=1,2,\dots,N} \omega_k. \tag{8}$$

Examples of operators belonging to the class in Assumption 5 are the *sparsifying compression operator* [22] and the *randomized quantizer* [18]. Let us focus on the latter operator, which can be briefly described as follows. Given an analog value $x \in \mathbb{R}^M$, its norm $\|x\|$ is assumed to be transmitted as virtually unquantized (i.e., represented with machine precision). Then, each entry $x_m$ of $x$ is compressed as follows. One bit is spent to represent the sign of $x_m$, whereas $r$ bits are employed to represent $|x_m|$ by randomly rounding it to one of the endpoints of the quantization interval it belongs to. We refer the reader to [18] for a detailed illustration of the pertinent algorithm. In summary, assuming, e.g., machine precision at 32 bits, the randomized quantizer in [18] requires $32 + M \times (r+1)$ bits for each vector to compress. Given $i_{\max}$ iterations, the expense of each agent is then:

$$\Big(32 + M \times (r+1)\Big) i_{\max}. \tag{9}$$

Moreover, it was shown that, as the bit-rate $r$ increases, the compression factor $\omega$ of the randomized quantizer in [18] scales as:

$$\omega \approx 2^{-2r}. \tag{10}$$

## 4. ACTC LEARNING DYNAMICS

We are now ready to illustrate the main result of this work, namely, that the the ACTC mean-square deviation approaches $O(\mu)$, i.e., for small $\mu$ each agent learns well *even in the presence of quantization errors and gradient noise*. The derivations can be found in [31]. They are demanding due to the nonlinear and coupled nature of the network dynamics, and are omitted for space constraints. To state Theorem 1 further ahead, we need to introduce a technical assumption on the gradient noise process $\nabla J_k(\boldsymbol{w}_{k,i-1}) - \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})$.

**Assumption 6** (Gradient Noise)**.** *For all $i > 0$, conditionally on the previous-step quantized iterates $\{\boldsymbol{q}_{\ell,i-1}\}_{\ell=1}^N$, the gradient noise has zero mean and obeys the bound, for some constants $\beta_k$ and $\sigma_k$:*

$$\begin{aligned}&\mathbb{E}\big[\|\nabla J_k(\boldsymbol{w}_{k,i-1}) - \widehat{\nabla J}_k(\boldsymbol{w}_{k,i-1})\|^2 \,\big|\{\boldsymbol{q}_{\ell,i-1}\}_{\ell=1}^N\big] \\ &\leq \beta_k^2 \|\boldsymbol{w}_{k,i-1} - w^\star\|^2 + \sigma_k^2,\end{aligned} \tag{11}$$

*where we recall that $w^\star$ is the minimizer of the cost function in* (4) *and that $\boldsymbol{w}_{k,i-1} = \sum_{\ell \in \mathcal{N}_k} a_{\ell k} \boldsymbol{q}_{\ell,i-1}$.* □

We continue by introducing the Jordan decomposition of the transposed combination matrix,

$$A^\top \triangleq V^{-1} J_{\rm tot} V, \quad J_{\rm tot} = \mathsf{diag}\{J_1, J_2, \dots, J_B\}. \tag{12}$$

In (12), $J_n = \lambda_n I_{L_n} + U_{L_n}$ is the $n$-th Jordan block associated with eigenvalue $\lambda_n$, where $U_{L_n}$ is a square matrix of size $L_n$ that has all zero entries, but for the superdiagonal, which has entries equal to 1. In view of Assumptions 2 and 3, matrix $A$ has a unique largest magnitude eigenvalue equal to 1, and we sort the remaining eigenvalues as $|\lambda_2| \geq |\lambda_3| \geq \ldots \geq |\lambda_B|$. It is convenient to introduce the block-diagonal matrices:

$$\Lambda \triangleq \mathsf{diag}\{\lambda_2 I_{L_2}, \lambda_3 I_{L_3}, \dots, \lambda_B I_{L_B}\}, \tag{13}$$

$$U \triangleq \mathsf{diag}\{U_{L_2}, U_{L_3}, \dots, U_{L_B}\}. \tag{14}$$

Let also

$$\Delta = \|V^{-1}\|^2 \max_{\substack{\ell=2,3,\dots,N \\ k=1,2,\dots,N}} |v_{\ell k}|^2\,\omega_k, \tag{15}$$

where $v_{\ell k}$ is the $(\ell, k)$-entry of $V$, and introduce the matrix

$$E = E_0 + 16\,\zeta^2\,\Delta\,\mathbb{1}_{N-1}\mathbb{1}_{N-1}^\top, \tag{16}$$

with

$$E_0 = \left((1-\zeta) I_{N-1} + \zeta\,\frac{\Lambda\Lambda^*}{|\lambda_2|}\right) + \frac{2\zeta}{1-|\lambda_2|}\,U. \tag{17}$$

The spectral radius of $E$ will be denoted by $\rho(E)$. Finally, let

$$a_n \triangleq \frac{2|\lambda_2|}{1-|\lambda_2|}\,\frac{1}{|\lambda_2| - |\lambda_n|^2}, \tag{18}$$

and

$$\gamma(A) \triangleq \sum_{n=2}^{B} \frac{|\lambda_2|}{|\lambda_2| - |\lambda_n|^2}\left(\frac{a_n^{L_n+1} - 1}{(a_n - 1)^2} + \frac{L_n + 1}{a_n - 1}\right). \tag{19}$$

**Theorem 1** (ACTC Learning Behavior)**.** *Let*

$$\rho_{\rm cen} \triangleq (1 - \mu\,\zeta\,\nu)^2, \quad \rho_{\rm net} = \rho(E) + \epsilon < 1, \tag{20}$$

*for some $\epsilon > 0$. If $\zeta < (16\Delta\gamma(A))^{-1}$, for sufficiently small step-size $\mu$ the evolution of the mean-square deviation of each agent $k$ can be cast in the form, for all $i > 0$:*

$$\begin{aligned}&\mathbb{E}\|\boldsymbol{w}_{k,i} - w^\star\|^2 \\ &\leq \underbrace{O(1)\,\rho_{\rm net}^i}_{\substack{\text{network convergence to}\\ \text{a coordinated evolution}}} + \underbrace{O(1)\,\rho_{\rm cen}^i}_{\substack{\text{coordinated}\\ \text{evolution}}} + \underbrace{O(\mu)\,\rho_{\rm cen}^{i/4}}_{\substack{\text{higher-order correction}\\ \text{relative to the transient phase}}} \\ &\quad \overbrace{+\,\mu\,\zeta\left(\underbrace{\frac{\sum_{\ell=1}^N \pi_\ell\,\alpha_\ell^2\sigma_\ell^2}{2\nu}}_{\text{uncompressed ACTC}} + \underbrace{c_q\,\Omega\,(1+\Omega)}_{\text{compression loss}}\right)}^{\text{steady-state error}} + O(\mu^{3/2}),\end{aligned} \tag{21}$$

*where $c_q > 0$ is a suitable constant independent of $\mu$ and $i$.* ■

Result (21) leads to a sharp description of adaptive diffusion with compressed communication, in terms of: $i$) two main *transient* terms that vanish as $i \to \infty$ with different rates; and $ii$) two *steady-state* terms, one corresponding to data shared with infinite precision, the other embodying the effect of compression.

— *Transient Phases*. The rate $\rho_{\rm net}$ depends only on the parameter $\zeta$, and on the network connectivity properties through the eigenspectrum of the combination matrix $A$. As a result, for sufficiently small $\mu$ we have that $\rho_{\rm cen} > \rho_{\rm net}$ and the associated transient (relative to the convergence of the agents to a coordinated evolution) dies out fast (Phase I). After this initial transient, a second transient dominates (Phase II), which is relative to the slower process of the agents' convergence to the steady-state. Remarkably, these two distinct phases have been shown to coexist also in adaptive learning over networks *without communication constraints* [5, 6, 11].

— *Compression Loss*. After transient Phase II, the following *upper bound* on the steady-state mean-square deviation holds:

$$\mathsf{MSD}_{\text{ACTC}} = \mu\,\zeta\left(\frac{\sum_{\ell=1}^{N}\pi_\ell\,\alpha_\ell^2\sigma_\ell^2}{2\nu} + c_q\,\Omega\,(1+\Omega)\right) + O(\mu^{3/2}). \tag{22}$$

First of all, the product $\mu\,\zeta$ stays fixed once we set a convergence rate $\rho_{\text{cen}}$. Then, for a given $\rho_{\text{cen}}$, the mean-square deviation is composed of two main terms: $i)$ the error, independent of the amount of compression, proportional to an average over the Perron weights $\{\pi_\ell\}$ of the gradient noise powers $\{\sigma_\ell^2\}$; $ii)$ the *compression loss*, which is an increasing function of the compression factor $\Omega$.

— *Comparison Against Classical ATC*. From (1) and (5), we see that the *uncompressed* ACTC (i.e., when $\boldsymbol{Q}_\ell(x) = x$) coincides with the classical ATC when $\zeta = 1$. Accordingly, by setting $\zeta = 1$ and $\Omega = 0$ in (22), we get an upper bound on the mean-square deviation of the ATC diffusion strategy:

$$\mathsf{MSD}_{\text{ATC}} = \mu\,\frac{\sum_{\ell=1}^{N}\pi_\ell\,\alpha_\ell^2\sigma_\ell^2}{2\nu} + O(\mu^{3/2}). \tag{23}$$

To compare (23) against (22), we need to set the same convergence rate, which is tantamount to setting the step-size $\mu$ in (23) and the product $\mu\,\zeta$ in (22) equal to the same value, yielding the following scaling law w.r.t. the compression loss factor $\Omega$:

$$\mathsf{MSD}_{\text{ACTC}} - \mathsf{MSD}_{\text{ATC}} \sim \Omega\,(1+\Omega). \tag{24}$$

For example, for the randomized quantizer described in Sec. 3.1, using (10) and (24), we conclude that, for sufficiently small step-sizes we have $\mathsf{MSD}_{\text{ACTC}} - \mathsf{MSD}_{\text{ATC}} \sim 2^{-2r_{\min}}$ (where $r_{\min}$ is the smallest resolution across the agents), which leads to the following insightful conclusion. The inference (i.e., estimation) error on the desired parameter $w^\star$ decouples into two terms: a constant *inference* error that corresponds to the performance achievable without compression *plus* a *reproduction* error that arises from data compression and vanishes exponentially fast with the bit-rate.

## 5. ILLUSTRATIVE EXAMPLE

As an application of the ACTC diffusion strategy, we consider the scenario where $N$ agents aim at solving a regression problem in a distributed way. Each agent $k$ observes a flow of streaming data $\boldsymbol{d}_{k,i} \in \mathbb{R}$ and regressors $\boldsymbol{u}_{k,i} \in \mathbb{R}^M$, which obey the model:

$$\boldsymbol{d}_{k,i} = \boldsymbol{u}_{k,i}^\top w^\star + \boldsymbol{v}_{k,i} \quad k = 1, 2, \dots, N, \tag{25}$$

where $w^\star \in \mathbb{R}^M$ is an unknown (deterministic) parameter vector and $\boldsymbol{v}_{k,i} \in \mathbb{R}$ acts as noise. The goal is to to estimate the unknown $w^\star$, which corresponds to the optimization problem:

$$\min_{w\in\mathbb{R}^M} \mathbb{E}\left[\sum_{k=1}^{N} p_k\left(\boldsymbol{d}_{k,i} - \boldsymbol{u}_{k,i}^\top w\right)^2\right]. \tag{26}$$

Processes $\{\boldsymbol{u}_{k,i}\}$ and $\{\boldsymbol{v}_{k,i}\}$ are independent over time and across agents. The noise variables are zero-mean Gaussian with variances different across agents, and chosen uniformly at random in the interval $(0.1, 0.5)$. The entries of $\boldsymbol{u}_{1,i}$ are independent standard Gaussian variables, while for agents $k = 2, 3, \dots, N$ the first $M-1$ entries of $\boldsymbol{u}_{k,i}$ are independent standard Gaussian variables, and the last two entries are equal. As a result, the regressor covariance matrices of these agents are singular, ensuring only convexity of their individual cost functions. The global cost function in (4) is strongly convex since the regressor covariance matrix of agent 1 is invertible.

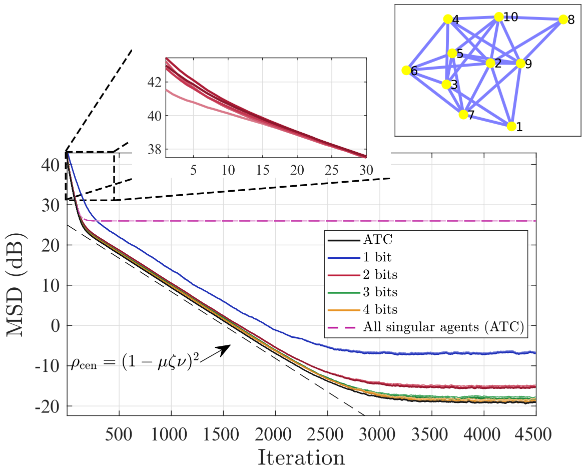

**Fig. 1**. Mean-square deviation of agents $3, 4, 5, 6, 7$ as a function of time, for different bit-rates. The simulation setting is described in Sec. 5. All errors are estimated by means of $10^2$ Monte Carlo runs.

In Fig. 1, we examine the learning performance of the ACTC diffusion strategy as a function of the iteration $i$, for different quantizers' resolutions. The regression problem has dimensionality $M = 50$. The network is made of $N = 10$ agents that interact over the topology displayed in the top-right panel, employing a left-stochastic policy defined according to the averaging rule [11]. We set $\zeta = 0.15$, and equal step-sizes $\mu_k = \mu = 6\times 10^{-2}$. All agents employ the randomized quantizers described in Sec. 3.1 and use the same number of bits $r$, ranging from 1 to 4.

The behavior observed in Fig. 1 matches the results in Theorem 1: $i)$ for all bit-rates, there is a transient governed by the predicted rate $\rho_{\text{cen}}$; $ii)$ higher-order discrepancies are absorbed into an initial, much faster, transient; $iii)$ the ACTC errors converge to different steady-state values that, yet for relatively low bit-rates, approach the performance of the ATC diffusion strategy. The magenta curve refers to the case where all agents are singular (here we used classical ATC to give them an advantage), and shows that the agents deliver poor performance. In contrast, when $N-1$ singular agents cooperate with a farsighted agent, the distributed information sharing drives them to the correct vector $w^\star$, even with compressed data.

It is useful to evaluate the savings achieved with the ACTC strategy. With reference to Fig. 1, for the time needed to enter reliably the steady state ($i_{\max} \approx 3000$), and using $r = 2$ bits, we get:

$$r_{\text{ACTC}} = (32 + 50 \times 3) \times 3000 = 546 \text{ kbit}. \tag{27}$$

For the plain ATC strategy, where each entry of the vector to be quantized is represented by 32 bits, we get:

$$r_{\text{ATC}} = 32 \times 50 \times 3000 = 4.8 \text{ Mbit}, \tag{28}$$

implying a remarkable gain of about one order of magnitude. This gain should be evaluated in relation to the loss in mean-square deviation. We see that we lose $\approx 4$ dB, which is definitely tolerable, especially in the light of the remarkable bit-rate savings.

## 6. REFERENCES

[1] U. A. Khan, W. U. Bajwa, A. Nedić, M. G. Rabbat, and A. H. Sayed, *Editors*, "Optimization for data-driven learning and control," *Proceedings of the IEEE*, vol. 108, no. 11, pp. 1863–1868, Nov. 2020.

[2] J. N. Tsitsiklis, D. P. Bertsekas, and M. Athans, "Distributed asynchronous deterministic and stochastic gradient optimization algorithms," *IEEE Trans. Autom. Control*, vol. 31, no. 9, pp. 803–812, Sep. 1986.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, "Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers," *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, 2010.

[4] A. H. Sayed, "Adaptive networks," *Proceedings of the IEEE*, vol. 102, no. 4, pp. 460–497, Apr. 2014.

[5] J. Chen and A. H. Sayed, "On the learning behavior of adaptive networks — part I: Transient analysis," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3487–3517, Jun. 2015.

[6] J. Chen and A. H. Sayed, "On the learning behavior of adaptive networks — part II: Performance analysis," *IEEE Trans. Inf. Theory*, vol. 61, no. 6, pp. 3518–3548, Jun. 2015.

[7] M. H. DeGroot, "Reaching a consensus," *J. Amer. Statist. Assoc.*, vol. 69, no. 345, pp. 118–121, 1974.

[8] L. Xiao and S. Boyd, "Fast linear iterations for distributed averaging," *Systems and Control Letters*, vol. 53, no. 1, pp. 65–78, Sep. 2004.

[9] S. Boyd, A. Ghosh, B. Prabhakar, and D. Shah, "Randomized gossip algorithms," *IEEE Trans. Inf. Theory*, vol. 52, no. 6, pp. 2508–2530, Jun. 2006.

[10] A. G. Dimakis, S. Kar, J. M. F. Moura, M. G. Rabbat, and A. Scaglione, "Gossip algorithms for distributed signal processing," *Proceedings of the IEEE*, vol. 98, no. 11, pp. 1847–1864, Nov. 2010.

[11] A. H. Sayed, "Adaptation, Learning, and Optimization over Networks," *Found. Trends Mach. Learn.*, vol. 7, no. 4-5, pp. 311–801, 2014.

[12] V. Matta and A. H. Sayed, "Estimation and detection over adaptive networks," in *Cooperative and Graph Signal Processing*, P. Djuric and C. Richard, Eds. Elsevier, 2018, pp. 69–106.

[13] B. Widrow, "A study of rough amplitude quantization by means of Nyquist sampling theory," *IRE Trans. on Circuit Theory*, vol. 3, no. 4, pp. 266–276, Dec. 1956.

[14] R. M. Gray and T. G. Stockham, "Dithered quantizers," *IEEE Trans. Inf. Theory*, vol. 39, no. 3, pp. 805–812, May 1993.

[15] Z.-Q. Luo, "Universal decentralized estimation in a bandwidth constrained sensor network," *IEEE Trans. Inf. Theory*, vol. 51, no. 6, pp. 2210–2219, Jun. 2005.

[16] J. B. Predd, S. B. Kulkarni, and H. V. Poor, "Distributed learning in wireless sensor networks," *IEEE Signal Process. Mag.*, vol. 23, no. 4, pp. 56–69, Jul. 2006.

[17] S. Marano, V. Matta, and P. Willett, "Nearest-neighbor distributed learning by ordered transmissions," *IEEE Trans. Signal Process.*, vol. 61, no. 21, pp. 5217–5230, Nov. 2013.

[18] D. Alistarh, D. Grubic, J. Z. Li, R. Tomioka, and M. Vojnovic, "QSGD: communication-efficient SGD via gradient quantization and encoding" in *Proc. Neural Information Processing Systems (NIPS)*, Long Beach, CA, USA, Dec. 2017, pp. 1707–1718.

[19] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Springer Science+Business Media, New York, 2001.

[20] F. Seide, H. Fu, J. Droppo, G. Li, and D. Yu, "1-Bit stochastic gradient descent and its application to data-parallel distributed training of speech DNNs," in *Proc. Conference of the International Speech Communication Association*, Singapore, Sep. 2014, pp. 1058–1062.

[21] J. Wu, W. Huang, J. Huang, and T. Zhang, "Error compensated quantized SGD and its applications to large-scale distributed optimization," in *Proc. International Conference on Machine Learning (ICML)*, Stockholm, Sweden, Jul. 2018, pp. 5235–5333.

[22] S. U. Stich, J-B. Cordonnier, and M. Jaggi, "Sparsified SGD with memory," in *Proc. Neural Information Processing Systems (NIPS)*, Montréal, Canada, Dec. 2018, pp. 4447–4458.

[23] S. P. Karimireddy, Q. Rebjock, S. U. Stich, and M. Jaggi, "Error feedback fixes SignSGD and other gradient compression schemes," in *Proc. International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, Jun. 2019, pp. 3252–3261.

[24] C.-Y. Lin, V. Kostina, and B. Hassibi, "Differentially quantized gradient descent," in *Proc. IEEE International Symposium on Information Theory (ISIT)*, Melbourne, Victoria, Australia, Jul. 2021, pp. 1200–1205.

[25] X. Zhao, S.-Y. Tu, and A. H. Sayed, "Diffusion adaptation over networks under imperfect information exchange and nonstationary data," in *IEEE Trans. Signal Process.*, vol. 60, no. 7, Apr. 2012, pp. 3460–3475.

[26] A. Koloskova, S. U. Stich, and M. Jaggi, "Decentralized stochastic optimization and gossip algorithms with compressed communication," in *Proc. International Conference on Machine Learning (ICML)*, Long Beach, CA, USA, Jun. 2019, pp. 3478–3487.

[27] D. Kovalev, A. Koloskova, M. Jaggi, P. Richtárik, and S. U. Stich, "A linearly convergent algorithm for decentralized optimization: sending less bits for free!," in *Proc. International Conference on Artificial Intelligence and Statistics (AISTATS)*, San Diego, CA, USA, Apr. 2021, pp. 4087–4095.

[28] J. Chen and A. H. Sayed, "Distributed Pareto optimization via diffusion strategies," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 2, pp. 205–220, Apr. 2013.

[29] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, "Exact diffusion for distributed optimization and learning – Part I: Algorithm development," in *IEEE Trans. Signal Process.*, vol. 67, no. 3, pp. 708–723, Feb. 2019.

[30] K. Yuan, B. Ying, X. Zhao, and A. H. Sayed, "Exact diffusion for distributed optimization and learning – Part II: Convergence analysis," in *IEEE Trans. Signal Process.*, vol. 67, no. 3, pp. 724–739, Feb. 2019.

[31] M. Carpentiero, V. Matta, and A. H. Sayed, "Distributed adaptive learning under communication constraints," *submitted for publication*, Nov. 2021. Available online as arXiv:2112.02129v1 [cs.LG].