A Feasible Algorithm for Distributed Resource Allocation with Noisy Communication and Computation

Albert Zi Wen Cheng ∗ Cédric Langbort ∗∗

∗ University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA
(e-mail: cheng73@illinois.edu).

∗∗ University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA
(e-mail: langbort@illinois.edu).

Abstract: We aim to apply a simple deterministic feasible resource allocation algorithm to wireless sensor networks. Since wireless sensor networks are inherently subject to computation and communication noises, we modify this deterministic algorithm into a stochastic algorithm and show that under suitable conditions, the stochastic algorithm converges to the optimal value almost surely.

Keywords: sensor networks, distributed control and estimation, multi-agent systems.

1. INTRODUCTION

Many algorithms have been proposed to distributively solve convex optimization problems in a network environment (see, e.g., (Johansson, 2008)). In this paper, we focus on the problem of distributed convex optimization under an equality constraint, which constitutes a standard model for resource allocation. In this context, it is often desirable for the equality constraint (which captures the fact that the amount of physically available resources is limited) to be satisfied at every iteration of the algorithm. This guarantees that the allocation obtained upon termination is always implementable, even if the nodes of the network are unable to run the algorithm to completion because of computational limitations or limited run-time requirements. An algorithm whose iterates satisfy the constraint at every time step is said to be “feasible”, since it generates a sequence of feasible points for the underlying optimization problem.

Dual decomposition algorithms, which introduce the Lagrange multiplier associated with the equality constraint as a coordination variable among decoupled subproblems solved at each node, are not feasible since they only guarantee that the constraint is satisfied asymptotically (Culioli and Cohen, 1990; Boyd and Vandenberghe, 2004). Primal decomposition algorithms are feasible, but require communication between all the nodes of the network if the equality constraint involves global coupling.

In contrast with these methods, Xiao and Boyd recently proposed a feasible distributed resource allocation algorithm for a noise-free network, which only requires communication between direct neighbors in the network (Xiao and Boyd, 2006). This algorithm uses a very simple mechanism to guarantee feasibility at every iteration, and thus constitutes an attractive starting point for more complex wireless networks problems, in which computation and communication noises cannot be avoided (Kurose and Simha, 1989; Giridhar and Kumar, 2005). Since this algorithm does not converge in the presence of noise, we propose a slight modification, based on tools and ideas from the field of stochastic gradient approximation (Robbins and Monro, 1951; Bottou, 1998), which results in a new algorithm, whose almost sure convergence to an optimal resource allocation can be established, in the presence of noise.

We start by reviewing some features of the Xiao-Boyd algorithm introduced in (Xiao and Boyd, 2006) in Section 2. We then propose our extension in Section 3. The effect of noise is analyzed in Section 4, which also contains our main convergence theorem. We illustrate the performance of the stochastic algorithm through a numerical example and discuss future work in Section 5. Finally, we conclude the paper in Section 6.

1.1 Problem Formulation

We consider a network of autonomous agents modeled as a strongly connected directed graph \((V, E)\), where \(V = \{1 \ldots n\}\) is the set of nodes, and \(E \subseteq V \times V\) denotes the edge set. Each edge \((i, j)\) is an ordered pair of distinct nodes. We define \(N_i = \{j|(i,j) \in E\}\) as the set of out neighbors of node \(i\). This network model constrains the way in which agents share information with each other, namely each agent can pass information to \(j\) only if \((i,j) \in E\).

In addition, we assume that each agent in the network is subjected to noises from computation and communication with other agents. To characterize these noises, let us define a probability space \((\Omega, \mathcal{F}, P)\), where \(\Omega\) is the set of outcomes, \(\mathcal{F}\) is the \(\sigma\)-algebra generated by the open sets of \(\Omega\), and \(P\) is the probability measure. Let \((\mathcal{F}(k))_{k \in \mathbb{N}}\) be an increasing family of sub-\(\sigma\)-algebras of \(\mathcal{F}\) such that \(\cup_{k \in \mathbb{N}} \mathcal{F}(k) = \mathcal{F}\).

The computation noise, \(w_m\), and the communication noise, \(v_{c_{j ightleftharpoons i}}\) are modeled as the following real valued random processes.
For simplicity, we define the random variables \( v_{m_i}(k) \) as
\[
[v_{m_i}(k)](\omega) = v_{m_i}(\omega, k) \quad \forall \omega \in \Omega
\]
and \( v_{c_j \rightarrow i}(k) \) as
\[
[v_{c_j \rightarrow i}(k)](\omega) = v_{c_j \rightarrow i}(\omega, k), \quad \forall \omega \in \Omega
\]
for all \( k \). These random variables are measurable with respect to \( \mathcal{F}(k) \) for all \( k \).

The computational noise and the communication noise have properties that result from our network model and our application. Each agent’s computational ability, and consequently computation noise, is a unique property of that specific agent because we are assuming that each agent is autonomous. Also, the noise associated with each communication channel is independent of all other channels in the network. Finally, the cause of computation noise is generally different from the cause of communication noise in application. We summarize these properties and make some standard assumptions in Assumption 1.

**Assumption 1.** We assume that

1. The random processes \( (v_{m_i}(k))_{k \in \mathbb{N}} \) and \( (v_{c_j \rightarrow i}(k))_{k \in \mathbb{N}} \) are both i.i.d.
2. \( \mathbb{E}[v_{m_i}(k)] = 0 \) and \( \mathbb{E}[v_{c_j \rightarrow i}(k)] = 0 \) for all \( i, j, k \) and \( i \neq j \).
3. \( v_{m_i}(k) \) and \( v_{c_j \rightarrow i}(l) \) are independent \( \forall i \neq j, \forall k, l \).
4. \( v_{c_j \rightarrow i}(k) \) and \( v_{c_q \rightarrow m}(l) \) are independent \( \forall (i, j) \neq (a, p), \forall k, l \).
5. \( v_{c_j \rightarrow i}(k) \) and \( v_{c_q \rightarrow m}(l) \) are independent \( \forall i \neq j, \forall k, l \).

A real variable \( x_i \in \mathbb{R} \) and a convex objective function \( f_i \) are associated with each node \( i \) in the network. Our goal is to solve the following problem taking into account the computation and communication noises of each agent:

\[
\min_{x_1, \ldots, x_n} \sum_{i=1}^{n} f_i(x_i) \quad \text{s.t.} \quad \sum_{i=1}^{n} x_i = c,
\]

for some constant \( c \in \mathbb{R} \). For simplicity, we define

\[
x = [x_1, x_2, \ldots, x_n]^t, \quad \hat{1} = [1, 1, \ldots, 1]^t, \quad F(x) = \sum_{i=1}^{n} f_i(x_i),
\]

with \(^t\) denoting the transpose of a vector. Our minimization problem then becomes

\[
\min_{x} F(x) \quad \text{s.t.} \quad \hat{1}^t x = c. \tag{1}
\]

We make the following assumptions about the objective function of each agent.

**Assumption 2.** We assume that for each \( i \)

1. \( f_i : \mathbb{R} \rightarrow \mathbb{R}^+ \) is a twice continuously differentiable non-negative function.
2. \( \exists u_i \in \mathbb{R}^+ \) such that \( \frac{\partial^2}{\partial x^2} f_i(x_i) \leq u_i \) for all \( x_i \).
3. \( f_i \) is a strictly convex function.
4. \( f_i \) is coercive, i.e., \( \lim_{|x_i| \rightarrow +\infty} f_i(x_i) = +\infty \).

### 1.2 Previous Work

The algorithm developed in this paper belongs to a more general class of stochastic gradient descent algorithms in (Tsitsiklis et al., 1986). Our algorithm is also related to algorithms in (Ram et al., 2008), (Ram et al., 2009), and (Sundar et al., 2010). In (Ram et al., 2008), the authors combine the incremental gradient algorithm (Nedic and Bertsekas, 2001) with Robbins Monro approximation method (Robbins and Monro, 1951) to solve a stochastic optimization problem. This algorithm requires a ring structure in the network to pass information between nodes. In (Ram et al., 2009), each node randomly selects a neighbor and averages it’s state with the selected neighbor. Then, each node adjusts the average along the negative direction of the gradient. The state is a scalar and is required to compute every agent’s objective function.

Finally, the algorithm in (Ram et al., 2008) updates the state with a sub-gradient evaluated at a weighted average of the states of its neighbors. This algorithm solves the problem where the entire state is needed to calculate each agent’s objective function.

Our algorithm differs from these related algorithms in that we assume each agent can compute their objective function using only their local state. Also, instead of using projection to satisfy the constraint at each iteration, our algorithm uses a simpler mechanism in choosing form of the \( W \) matrix. Finally, our algorithm does not require the network to have a specific communication structure such as the ring structure in (Ram et al., 2008).

### 2. PRELIMINARIES

We begin by reviewing the main properties of the following algorithm introduced in (Xiao and Boyd, 2006)

\[
x(k+1) = x(k) - W \nabla F(x(k)), \tag{2}
\]

\[
\nabla F(x) = \left[ \frac{d}{dx_1} f_1(x_1) \cdots \frac{d}{dx_n} f_n(x_n) \right]^t. \tag{3}
\]

The elements in the matrix \( W = [w_{ij}] \) are such that \( w_{ij} = 0 \) \( \forall j \notin N_i, \ w_{ij} \neq 0 \ \forall j \in N_i, \ w_{ii} \neq 0 \ \forall i \). In addition, \( W \) is assumed to satisfy\( \hat{1}^t W = 0, \ W \hat{1} = 0. \tag{4} \)

Property (4) is used to ensure that algorithm (2) satisfies the feasibility condition

\[
\hat{1}^t x(k+1) = \hat{1}^t x(k) - \hat{1}^t W \nabla F(x(k)) = \hat{1}^t x(k) = c. \tag{5}
\]

Also, property (4) guarantees that the optimal solution of (1), \( x^* \), is a stationary point of algorithm (2) since

\[
x^* = x^* - W(\lambda^* \hat{1}) = x^*. \tag{6}
\]

The fact that \( \nabla F(x^*) = \lambda^* \hat{1} \) follows from the KKT conditions (Boyd and Vandenberge, 2004)

\[
\nabla F(x^*) = \lambda^* \hat{1}, \quad \text{for a unique } \lambda^* \in \mathbb{R}, \tag{7}
\]

\[
\hat{1}^t x^* = c. \tag{8}
\]

Under these assumptions on \( W \) and an additional assumption on the eigenvalues of \( W \), (Xiao and Boyd, 2006) proved that algorithm (2) converges to the optimal solution of (1) in a directed network with no noise.

### 3. ALGORITHM

We cannot directly apply algorithm (2) to problem (1) in our network because the implicit assumption of perfect gradient information in algorithm (2) is not valid in a network with computation and communication noise.
Taking inspiration from algorithm (2), we propose the following extension for each agent $i$ in our network
\[ x_i(k+1) = \ldots = \frac{d}{dx_i} f_i(x_i(k) + v_m(k)) \]
\[ + \sum_{i \in N_j, j \neq i} w_{ij} \left( \frac{d}{dx_j} f_j(x_j) + v_m(j) + v_{c_{j,i}}(k) \right) \],

(9)

with $x_i(0) = x_i^0$, where $x_i^0$ is deterministic and feasible. The sequence $\{\gamma(k)\}_{k \in \mathbb{N}}$ is monotonically decreasing and
\[ \gamma(k) > 0, \quad \sum_{k=1}^{\infty} \gamma(k) = \infty, \quad \sum_{k=1}^{\infty} \gamma(k)^2 < \infty, \quad \forall k. \] (10)

Such a step is standard in Robbins-Monro type stochastic gradient algorithms (Robbins and Monro, 1951) and is a typical way to ensure convergence. We rewrite algorithm (9) with simplified notation as
\[ v_m(k) = \left[ v_m(n_1) \cdots v_m(n_k) \right]', \quad r(k) = \nabla F(x(k)) + v_m(k), \]
\[ v_{c}(k) = \left[ \sum_{i \in N_j} w_{ij} v_{c_{j,i}}(k) \right]' \quad \text{and} \quad x(k+1) = x(k) - \gamma(k) (W r(k) + v_c(k)). \] (11)

Note that if $i \neq j$, the random variables $r_i(k)$ and $r_j(k)$ can be correlated if the network graph contains a node $l$ with the property that there exist directed paths of length less than $k$ between $l$ and $i$, and $l$ and $j$.

Let us define
\[ \Gamma(k) = \left[ x(1) \cdots x(k), v_m(1) \cdots v_m(k-1) \right]' \]
and denote the expectation conditioned on $\Gamma(k)$ as $\mathbb{E}_k \{ \cdot \} = \mathbb{E}\{|\Gamma(k)| \}$. We note from Assumption 1 and the definition of $\Gamma(k)$ that
\[ \mathbb{E}_k \{ \nabla F(x(k)) + v_m(k) \} = \nabla F(x(k)). \] (12)

In the next section, we analyze the effects of noise and show that under certain conditions, algorithm (11) converges almost surely to the optimal value of problem (1).

4. PROPERTIES

4.1 Noise Effects

Ideally, we would like algorithm (11) to be feasible in the sense that constraint (5) should hold. While this is not possible because of the noise, assuming $\tilde{W} = 0$ allows us to prove that algorithm (11) retains a property analogous to equation (5), namely:
\[ \mathbb{E}\{\tilde{\Gamma} x(k+1)\} = \mathbb{E}\{\tilde{\Gamma} x(k)\} + \gamma(k) \mathbb{E}\{\tilde{\Gamma} v_c(k)\} \]
\[ + \gamma(k) \tilde{W} \mathbb{E}\{\nabla F(x(k))\} = \mathbb{E}\{\tilde{\Gamma} x(k)\} \]. (13)

Furthermore, provided that one can establish the convergence of sequence $\{x(k)\}$ to $\hat{x}$ (a point we will tackle shortly), it can also be shown that the variance of $\tilde{\Gamma} x$ is bounded if there exists $\beta \in \mathbb{R}^+$ such that $Var(v_c(k)) \leq \beta$ for all $i$. From Assumption 1, we have
\[ \tilde{\Gamma} x(k+1) = Var(\tilde{\Gamma} x(k) + \gamma(k) \tilde{\Gamma} v_c(k)) \]
\[ = Var(\tilde{\Gamma} x(k)) + \gamma(k)^2 Var(\tilde{\Gamma} v_c(k)) \leq n \beta \sum_{q=1}^{\infty} \gamma(q)^2, \] (14)

where $n$ is the number of agents in the network. Using the assumption in (10) that $\{\gamma(k)\}$ is square-summable yields
\[ \lim_{k \to \infty} Var(\tilde{\Gamma} x(k)) = Var(\tilde{\Gamma} \hat{x}) \leq n \beta \sum_{q=1}^{\infty} \gamma(q)^2 < \infty. \] (15)

From the upper bound on the variance of the equality constraint (14), we see that the variance of the equality constraint at each iteration is a function of the communication noise only. It follows that algorithm (11) is feasible when computation noise is the only one present.

4.2 Convergence Analysis

We have shown the effects of computation and communication noise on algorithm (11). Next, we analyze the convergence of algorithm (11). Our analysis results in Theorem 4. Its proof relies on the Quasi-Martingale Convergence Theorem, which is a classical tool for the analysis of stochastic gradient algorithms (Bottou, 1998; Culioli and Cohen, 1990). We state this theorem as Lemma 3.

Lemma 3. Consider the probability space introduced in Section 1.1. Let $\{Y(k)\}_{k \in \mathbb{N}}$ be a real valued stochastic process such that $Y(k) = \mathcal{F}(k)$ measurable for every $k$. For every $k$, let the event $G(k)$ be defined as
\[ G(k) := \mathbb{E}\{Y(k+1) - Y(k)|\mathcal{F}(k)\} > 0 \]
and define the indicator function $I_{G(k)}$ as
\[ I_{G(k)} = \begin{cases} 1 & \text{if } G(k) \text{ is true} \\ 0 & \text{otherwise} \end{cases} \] (16)

Finally, let us assume that
\[ \sum_{k=1}^{\infty} \mathbb{E}\{I_{G(k)} \cdot (Y(k+1) - Y(k))\} < \infty \] (17)
\[ \liminf_{k \to \infty} \mathbb{E}\{Y(k)\} > -\infty \] (18)
\[ \sup_{k \to \infty} \mathbb{E}\{Y(k)^{-}\} < \infty, \] (19)

where $Y(k)^{-} = -\min\{0, Y(k)\}$. Then $\{Y(k)\}_{k \in \mathbb{N}}$ converges to an integrable random variable $Y_{\infty}$ almost surely.

Proof. See page 49-51 of (Métivier, 1982).

We can now proceed to state and prove Theorem 4.

Theorem 4. Let Assumption 1 and Assumption 2 hold, and let $W$ be a square matrix with the following properties:
\[ \tilde{W} = 0, \quad W^T = W, \] (20)
\[ \lambda_m > 0, \] (21)

where $\lambda_m$ is defined as the second smallest eigenvalue of $W$. Assume that there exist $\alpha_1, \alpha_2, \alpha_3, \beta \in \mathbb{R}^+$ such that
\[ Var(r_i(k)) \leq \alpha_1, \quad \mathbb{E}\{r_i(k)^2\} \leq \alpha_2^2, \quad \mathbb{E}\{r_i(k)^2\} \leq \alpha_3, \] (22)

for all $i, k$. Here, $r_i(k)$ is the $i$-th element of vector $r(k)$, and
\[ Var(v_{c_i}(k)) < \beta \quad \forall i, k. \] (23)

Then $F(x(k))$ converges almost surely to some random variable $F_\infty$. If there is no communication noise, then $F_\infty = F^*$, the optimal value of problem (1). In addition, $x(k)$ converges to $x^*$ almost surely, with $x^*$ being unique optimal point of problem (1).

Proof. From Taylor’s Theorem of Remainders, there exists $z_i(k) \in (x_i(k), x_i(k+1))$ for every $i$ and $k$ such that
\[ f_i(x_i(k+1)) = f_i(x_i(k)) + \frac{d}{dx_i} f_i(x_i(k))(x_i(k+1) - x_i(k)) \]
\[ + \frac{1}{2} \frac{d^2}{dx_i^2} f_i(z_i(k))(x_i(k+1) - x_i(k))^2. \] (24)
Let $\Delta x(k) = x(k+1) - x(k)$. Summing over $i$, we have
\[ F(x(k+1)) = F(x(k)) + \nabla F(x(k))'\Delta x(k) + (1/2)\Delta x(k)'\nabla^2 F(z(k))\Delta x(k), \]
where $\nabla F$ is defined in (3), $\nabla^2 F$ is the (diagonal) Hessian matrix of $F$, and $z(k) = [z_1(k) \cdots z_n(k)]'$. From Assumption 2, there exists a diagonal matrix $U$ such that $U \geq \nabla^2 F(x, \forall x)$. Since $\Delta x(k) = -\gamma(k)(Wx(k) + v_c(k))$ and taking the expectation of (24), we obtain
\[ \mathbb{E}[F(x(k+1))] \leq \mathbb{E}[F(x(k))] - \gamma(k)(\mathbb{E}[\nabla^2 F(x(k))Wx(k)] + (1/2)\gamma(k)^2(\mathbb{E}[\nabla F(x(k))'UW\nabla F(x(k))] + 2\mathbb{E}[r(k)'UWv_c(k)] + \mathbb{E}[v_c(k)'Uv_c(k)]). \] 
For the second term of (25), using equation (12), we have
\[ \mathbb{E}[\nabla F(x(k))'UWx(k)] = \mathbb{E}[\hat{E}_k(\nabla F(x(k))')W\hat{E}_k{r(k)}] = \mathbb{E}[\nabla F(x(k))'/UW\nabla F(x(k))]. \] 
Let us orthogonally decompose $\nabla F(x(k))$ as $\nabla F(x(k)) = p(k)+c(k)$, where $p(k)$ denotes the component of $\nabla F(x(k))$ parallel to the vector 1, and $c(k)$ denotes the component in the hyperplane perpendicular to 1. Specifically, we define $e(k) = \nabla F(x(k)) - (1/\hat{1})\nabla F(x(k))$. (27)
Using (27) with assumptions (20) and (21) on matrix $W$, we can rewrite (26) as
\[ \mathbb{E}[\nabla F(x(k))'/UW\nabla F(x(k)) = \mathbb{E}[e(k)'W\nabla F(x(k)]) \geq \lambda_m \mathbb{E}[e(k)'c(k)$. Let us define $||\cdot||$ as the Euclidean norm of a vector. Since $\mathbb{E}[e(k)'c(k)] \geq ||\mathbb{E}[e(k)]||^2$, we have the inequality
\[ \mathbb{E}[\nabla F(x(k))'/UW\nabla F(x(k)) \geq \lambda_m ||\mathbb{E}[e(k)]||^2. \] 
In the third and fifth terms of inequality (25), $v_c(k)$ is independent of $x_i(k)$ for any $i, j \in \{1, \ldots, n\}$ by algorithm (11) and Assumption 1. Since $v_c(k)$ is independent of $v_c(k)$, $r(k)$ is independent of $v_c(k)$ as well. Therefore,
\[ \mathbb{E}[\nabla F(x(k))'v_c(k)] = \mathbb{E}[\nabla F(x(k))'e(k)] = 0, \]
\[ \mathbb{E}[r(k)'UWv_c(k)] = tr(WU\mathbb{E}[v_c(k)r(k)]) = 0, \]
\[ tr(WU\mathbb{E}[e(k)]'r(k)) = 0, \]
(30)
where $tr(.)$ denotes the trace of a matrix.
To analyze the fourth term in (25), let us define $Cov(.)$ as the covariance function. From assumption (22), we have
\[ |Cov(r_i(k), r_j(k))| \leq \sqrt{Var(r_i(k))Var(r_j(k))} \leq \alpha_1, \]
for all $i, j$, with $i \neq j$. Hence
\[ |Cov(r_i(k), r_j(k))| + |Cov(r_i(k), r_j(k))| \leq \alpha_1 + \alpha_2^2 \forall i, j, i \neq j. \] 
(31)
Let $\alpha_1 \in \mathbb{R}^+$ be such that
\[ tr(WU'W)' < \alpha_1. \] 
(32)
By assumption (22) and the upper bound inequality (31), there exists $\alpha_2 \in \mathbb{R}^+$ such that for $\alpha = n(n-1)$,
\[ tr(\mathbb{E}[r(k)'r(k)']) < n\alpha_2 + \alpha_1 + \alpha_2^2 < \alpha_2. \] 
(33)
Because $W'UW \geq 0$, we see that $\mathbb{E}[r(k)'W'UW\nabla F(x(k))] \geq 0$. If we define $\hat{\alpha} = \max(\alpha_1, \alpha_2)$, we can use the inequalities (32) and (33), the definition of $\hat{\alpha}$, and Cauchy-Schwarz inequality to upper bound the fourth term as
\[ \mathbb{E}[r(k)'W'UW\nabla F(x(k))] = tr(WU'W\mathbb{E}[r(k)'r(k)']) \leq \sqrt{tr(WU'W^2)tr(\mathbb{E}[r(k)'r(k)'])} \leq \sqrt{tr(WU'W^2)tr(\mathbb{E}[r(k)'r(k)'])} < \hat{\alpha}. \] 
(34)
Finally, let $\sigma$ be the largest element in $U$. Then, using assumption (23) and Assumption 1, we upper bound the last term of expectation inequality (25) as
\[ \mathbb{E}[v_c(k)'Uv_c(k)] \leq \sigma \sum_{i=1}^n Var(v_c(k)) \leq n\sigma. \] 
(35)
Combining the equalities (29) and (30) with inequalities (28), (34) and (35), and let $\Delta F(x(k)) = F(x(k+1)) - F(x(k))$ and $\kappa = (1/2)\hat{\alpha} + (1/2)n\sigma$, we conclude that
\[ \mathbb{E}[\Delta F(x(k))] \leq -\gamma(k)\lambda_m ||\mathbb{E}[e(k)]||^2 + \kappa \gamma(k)^2 \leq \kappa \gamma(k)^2. \] 
(36)
Now, we introduce the event $H(k)$ as
\[ H(k) := \{\mathbb{E}[\{F(x(k+1)) - F(x(k))\}r(k)] > 0\}, \]
and it’s indicator function as $I_{H(k)}$, similar to (16). Then, the upper bound on $\mathbb{E}[\Delta F(x(k))]$ in (36) yields
\[ \sum_{k=1}^\infty \mathbb{E}[I_{H(k)} \cdot \Delta F(x(k))] \leq \kappa \sum_{k=1}^\infty \gamma(k)^2 < \infty, \] 
(37)
Taking the infinite sum of (37) on both sides yields
\[ \sum_{k=1}^\infty \mathbb{E}[I_{H(k)} \cdot \Delta F(x(k))] \leq \kappa \sum_{k=1}^\infty \gamma(k)^2 < \infty, \] 
(38)
where the last inequality follows from the square summable property of $\gamma(k)$ in (10). We see from the upper bound of the infinite summation in (38) that assumption (17) in Lemma 3 is satisfied. Also, since $F$ is a non-negative function by Assumption 2, conditions (18) and (19) of Lemma 3 are satisfied. Consequently, we conclude by Lemma 3 that the process $F(x(k))$ converges almost surely to some random variable $F_{\infty}$.

With no communication noise, inequality (24) becomes
\[ F(x(k+1)) = F(x(k)) + \gamma(k)\nabla F(x(k))'Wx(k) + (1/2)\gamma(k)^2W'UW\nabla F(x(k)). \] 
(39)
We take the expectation of (39) conditioned on $1_{H(k)}$ and use $\hat{E}_k[.]$ as in equation (12). We then obtain
\[ \hat{E}_k[\Delta F(x(k))] \leq -\gamma(k)\nabla F(x(k))'W\hat{E}_k{r(k)} + (1/2)\gamma(k)^2\hat{E}_k{r(k)'W'UW\nabla F(x(k))}. \] 
(40)
From equation (12), using definition (27) of $e(k)$ and assumptions (20) and (21) on matrix $W$, we have
\[ \nabla F(x(k))'W\hat{E}_k{r(k)} = e(k)'W\hat{E}_k{r(k)} \geq \lambda_m ||e(k)||^2. \] 
(41)
From the upper bound on $\mathbb{E}[r(k)'W'UW\nabla F(x(k))]$ in (34), we concluded that there exists $\hat{\alpha}$ such that
\[ \hat{E}_k{r(k)'W'UW\nabla F(x(k))} \leq \hat{\alpha}. \] 
(42)
From (41) and (42), we bound the inequality (40) as
\[ \hat{E}_k[\Delta F(x(k))] \leq -\lambda_m e(k)'e(k) + \hat{\alpha}. \] 
(43)
Since $F(x(k))$ converges almost surely, we conclude that
\[ \lambda_m \sum_{k=1}^\infty ||e(k)||^2 < \infty. \] 
(44)
because $||\gamma(k)||$ is square-summable. This implies that there exists a set of cluster points for the series $e(k)$ and a corresponding set of cluster points for $x(k)$. Furthermore, since $\gamma(k)||e(k)||^2 \geq 0$, it follows that $\liminf_{k \rightarrow \infty} ||e(k)||^2 = 0$ almost surely.
Let $\{e(\phi(k))\}$ be a subsequence of $\{e(k)\}$ that converges to zero almost surely. Let $\{x(\phi(k))\}$ be the corresponding
subsequence extracted from \( \{x(k)\} \). Since \( F(x(k)) \) converges almost surely to \( F_\infty \), \( \lim_{k \to \infty} F(x(\phi(k))) = F_\infty \) as well, which implies that the sequence \( \{F(x(\phi(k)))\} \) is bounded. By the positivity of \( f_i \), \( f_i(x_i, \phi(k)) \) is bounded for all \( k \) and \( i \). By the coercivity of each \( f_i \), each sequence \( \{x_i(\phi(k))\} \) is bounded for all \( k \) and, hence, admits a convergent subsequence. Then, using a diagonal argument, we can find an increasing map \( \psi : \mathbb{N} \to \mathbb{N} \) such that \( \{x_i(\psi(\phi(k)))\} \) converges for all \( i \), i.e., such that the vector-valued sequence \( \{x(\psi(\phi(k)))\} \) converges to some \( \hat{x} \).

By the convergence of \( \{\phi(k)\} \) there exists \( \lambda^* \in \mathbb{R}^+ \) such that \( \lambda^* \hat{x} = \lim_{k \to \infty} \nabla F(x(\psi(\phi(k)))) \) almost surely. Then, by continuity of \( \nabla F \) from Assumption 2, we have

\[
\lambda^* \hat{x} = \nabla F(\hat{x}) \text{ almost surely.} \tag{44}
\]

In other words, \( \hat{x} \) satisfies the KKT conditions. But, since the domain of \( F \) is open and we only have a linear equality constraint, (1) satisfies the Slater’s Condition, i.e., the KKT conditions are necessary and sufficient. As a result, (44) implies that \( \hat{x} \) is the unique minimum of \( F \) and, thus, \( F(\hat{x}) = F^* \).

Because \( F(x(k)) \) converges to \( F_\infty \) almost surely, every subsequence of \( \{F(x(k))\} \) must also converge to the same \( F_\infty \) almost surely. Since we have just showed that one of the subsequences of \( \{F(x(k))\} \) converges to \( F^* \), we conclude that \( F_\infty = F^* \). From strict convexity of \( F \), the \( x^* \) associated with \( F^* \) is also the unique global minimum. Therefore, every subsequence of \( x(k) \) converges to the same point \( x^* \) almost surely, and it follows that \( x(k) \) converges to \( x^* \) almost surely. \( \square \)

We end by noting that the results of Theorem 4 are still valid if we consider \( r(k) \) as a more general function of the form \( r(k) = [r_1(x(k), v_{m_1}(k)), \ldots, r_n(x_n(k), v_{m_n}(k))] \) with the property \( \mathbb{E}_k \{r(k)\} = \nabla F(x(k)) \) for all \( k \).

5. NUMERICAL SIMULATION

5.1 Simulation Setup

We compare the performance of algorithm (11) with algorithm (2) for agents with the same objective functions as those in the numerical example of (Xiao and Boyd, 2006). We consider a 3-regular strongly connected directed graph with \( n = 6 \) agents (instead of \( n = 20 \) as in (Xiao and Boyd, 2006) for the sake of simulation run time). The topology of the network can be inferred from the structure of the symmetric weight matrix

\[
W = \frac{1}{25} \begin{bmatrix}
3 & -1 & 0 & -1 & 0 & -1 \\
-1 & 3 & -1 & 0 & -1 & 0 \\
0 & -1 & 3 & -1 & 0 & -1 \\
-1 & 0 & -1 & 3 & -1 & 0 \\
0 & -1 & 0 & -1 & 3 & -1 \\
-1 & 0 & -1 & 0 & -1 & 3
\end{bmatrix}.
\]

For the simulation, the following family of non-negative functions is used:

\[
f_i(x_i)= \frac{1}{2}a_i(x_i-c_i)^2 + \log(1+e^{b_i(x_i-d_i)}), \tag{45}
\]

for all \( i \), with coefficients \( a_i, b_i, c_i, d_i \) generated randomly according to the uniform distribution on intervals \([0,2], [-2,2], [-10,10], [-10,10] \) respectively. This family of functions has the second derivative

\[
d^2 F_i(x_i) = a_i + \frac{b_i^2 e^{b_i(x_i-d_i)}}{(1+e^{b_i(x_i-d_i)})^2} > 0 \quad \forall x_i
\]

with the upper bounds \( a_i + \frac{1}{4}b_i^2 \).

Let us define each component of vector \( r(k) \) as

\[
r_i(k) = \frac{d}{dx_i} f_i(x_i) + v_m(k).
\]

The values for \( v_{m_1}(k) \) and \( v_{m_2}(k) \) are generated independently by a Gaussian pseudo-random number generator with a mean of 0 and a variance of 3. With no loss of generality, we assume that the resource constraint takes the form \( \sum_{i=1}^n x_i = 0 \) and choose the initial conditions \( x(0) = 0.1 \epsilon [1 -1 -1 -1 1 -1] \).

In the remainder of this section, we will say that “\( F(x(k)) \)” converges to a value with \( \epsilon \) if \( F(x(k)) \) satisfies

\[
|f_i(x_i(k+1)) - f_i(x_i(k))| < \epsilon, \quad \text{for all } i. \tag{46}
\]

This does not necessarily mean that \( x(k) \) itself converges.

5.2 Simulation Results

We ran multiple trials to observe the performance of algorithm (11) over a sufficiently large sample. Using a pseudo-random number generator in MATLAB, we generated the following vectors of coefficients

\[
a = [0.4518, 0.6222, 1.8098, 0.5161, 1.2057, 0.5934],
\]

\[
b = [-1.3172, 1.6935, 1.9190, -0.3651, 0.8449, -0.7249],
\]

\[
c = [-5.4467, 1.3959, 1.2226, -1.8979, 5.5651, 1.5167],
\]

\[
d = [-1.2860, 6.3037, 7.7776, 4.7558, 7.6516, -0.1572].
\]

For each trial, we first applied algorithm (2) to the objective functions in (45) until algorithm (2) converged to \( \hat{x}^* \) with \( \epsilon = 0.001 \). This convergence is guaranteed in (Xiao and Boyd, 2006). We set \( \hat{x}^* = F^* \) as the benchmark for the performance of algorithm (11). Then, we applied algorithm (11) to the objective functions in (45) until \( F(x(k)) \) converged to a value \( F_\infty \) with \( \epsilon = 0.001 \). We then computed the error \( F_\infty - F^* \).

![Fig. 1. The error from F* with γ(k) = \frac{12}{k} and no communication noise.](image-url)

We first simulated the case with no communication error. Since every function \( f_i \) in (45) is strictly convex, we expect algorithm (11) to converge to \( x^* \) almost surely by Theorem 4. In this simulation, we used the step \( \gamma(k) = \frac{12}{k} \). The result is plotted in Figure 1. The empirical mean of the error of algorithm (11) for 1000 trials is \( 1.03 \times 10^{-2} \), while its empirical variance is \( 5.2629 \times 10^{-5} \). Note that since \( x(k) \) is feasible for all \( k \) because of the absence of communication noise, \( F(x(k)) \geq F^* \) for all \( k \). In particular, \( F_\infty \geq F^* \). This is shown in Figure 1.
In the presence of communication noise, we ran 10000 trials with the step \( \gamma(k) = \frac{1}{\sqrt{k}} \). According to Theorem 4, we expect convergence of \( F(x(k)) \) to some random variable \( F_\infty \) in this case, but do not expect that \( F_\infty \) equals \( F^* \). We also expect the empirical mean of \( \hat{1}' \hat{x} \) to be small. Results are plotted in Figure 2. The empirical mean of the error of algorithm (11) for this simulation is 13.0261, while the variance is 205.5396. We also see that the mean of \( \hat{1}' \hat{x} \) is 0.0045277 with a variance of 42.9357.

In contrast with our observations on Figure 1, when communication noise is active, the feasibility condition only holds in expectation at every \( k \). Hence on a particular realization, \( x(k) \) may not be feasible for problem (1), and thus \( F_\infty < F^* \) is possible. This can be seen on Figure 2.

5.3 Discussion and Future Works

From the simulation, the performance of algorithm (11) coincides with theory. Algorithm (11) performs well without communication noise when compared to algorithm (2). However, with communication error, we cannot guarantee that the algorithm reaches the optimum.

The error from optimum in the case of no communication errors can be attributed to a couple of factors. If the magnitude of the \( \epsilon \) in definition of (46) is smaller, then the error of (11) decreases but the runtime of the simulation increases. Also, the fact that the noise is only pseudo-random may play a role in the deviation of the performance of algorithm (11) from algorithm (2).

Empirically, the algorithm’s speed of convergence and accuracy is heavily influenced by the step size. The step sizes in the simulation was determined by trial and error to obtain the best accuracy while achieving a relatively fast convergence rate. Also, the step size affects the communication noise since we multiply the communication noise by the step size in the algorithm. Therefore, we aim to find a rule for choosing the step size that optimizes the performance of algorithm (11) in future works.

6. CONCLUSION

We considered the problem of distributed resource allocation on a wireless sensor network. We extended the weighted averaging algorithm of Xiao and Boyd (Xiao and Boyd, 2006) to take into account computation and communication noise. We proved that, under suitable assumptions on the noise and with a similar choice of weighting matrix \( W \) as in (Xiao and Boyd, 2006), our algorithm converges almost surely to the optimal value of the optimization problem. Our next step will be to establish the convergence rate and finding a proper rule for choosing the step size.

ACKNOWLEDGEMENTS

The authors are grateful for helpful discussions with Prof. Mikael Johansson during a visit to KTH. We also thank Prof. Karl H. Johansson for hosting us during our visit.

REFERENCES


tion. Cambridge Univ Pr.


