5-17-2024

Learning Proximal Operators with Gaussian Process and Adaptive Quantization in Distributed Optimization

Aldo Duarte Vera Tudela
Louisiana State University and Agricultural and Mechanical College

LEARNING PROXIMAL OPERATORS WITH GAUSSIAN PROCESS AND ADAPTIVE QUANTIZATION IN DISTRIBUTED OPTIMIZATION

A Dissertation

Submitted to the Graduate Faculty of the
Louisiana State University and
Agricultural and Mechanical College
in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

in

The Department of Electrical Engineering & Computer Science

by
Aldo Duarte Vera Tudela
B.S., Pontifical Catholic University, Peru, 2012
M.S., Louisiana State University, 2018
August 2024

Acknowledgments

This dissertation would not be possible without several contributions. First and foremost, I would like to express my sincere gratitude to my advisor, Dr. Shuangqing Wei for his invaluable and insightful guidance and patience throughout the course of my study at Louisiana State University. I am grateful for his generous support through ups and downs on both academic and personal grounds for so many years.

I want to express my special gratitude to the members of my PhD committee, namely Dean's Representative Dr. Seungwon Yang from the School of Information Studies and Center for Computation and Technology, LSU, Dr. Morteza Naraghi-Pour, Department of Electrical and Computer Engineering, LSU, Dr. Xiangyu Meng, Department of Electrical and Computer Engineering, LSU, and Dr. Hongchao Zhang, Department of Mathematics, LSU for patiently attending my exams and providing me with their constructive comments and suggestions.

In addition, this dissertation is dedicated to all my family members, especially my parents, and my grand-parents for their support and encouragement throughout my academic life.

I am thankful to my girlfriend Paula Castillo for all her support during this journey. I would not have made it without you!

Finally, I want to thank all the wonderful people I met throughout my years at LSU. Each of you made this journey more enjoyable and I will cherish all the beautiful memories we shared throughout my life.

Table of Contents

List of Tables

List of Figures

Abstract

In networks consisting of agents communicating with a central coordinator and working together to solve a global optimization problem in a distributed manner, the agents are often required to solve private proximal minimization subproblems. Such a setting often requires a further decomposition method to solve the global distributed problem, resulting in extensive communication overhead. In networks where communication is expensive, it is crucial to reduce the communication overhead of the distributed optimization scheme. Integrating Gaussian processes (GP) as a learning component to the Alternating Direction Method of Multipliers (ADMM) has proven effective in learning each agent's local proximal operator to reduce the required communication exchange. In this work, we propose to combine this learning method with adaptive uniform quantization in a hybrid approach that can achieve further communication reduction when solving a distributed optimization problem with ADMM. This adaptive quantization first considers setting the mid-value and window length according to the mean and covariance given by GP. In a later stage of our study, this adaptation is extended to also consider the variation of the quantization bit resolution. In addition, a convergence analysis of this setting is derived, leading to convergence conditions and error bounds in the cases where convergence cannot be formally proven. Furthermore, we study the impact of the communication decision-making of the coordinator, leading to the proposition of several query strategies using the agent's uncertainty measures given by the regression process. Extensive numerical experiments of a distributed sharing problem with quadratic cost functions for the agents have been conducted throughout this study. The results have demonstrated that the various algorithms proposed have successfully achieved their primary goal of minimizing the over-

all communication overhead while ensuring that the global solutions maintain satisfactory levels of accuracy. The favorable accuracy observed in the numerical experiments is consistent with the findings of the derived convergence analysis. In instances where convergence proof is lacking, we have shown that the overall ADMM residual remains bounded by a diminishing threshold. This implies that we can anticipate our algorithmic solutions to closely approximate the actual solution, thus validating the reliability of our approaches.

Chapter 1. Introduction

In a distributed optimization framework where a group of agents is linked to a central coordinator, the optimization process typically involves agents tackling individual local sub-problems privately while maintaining frequent data exchanges with the coordinator. Many of these schemes are based on agents who solve *proximal minimization problems* [2] as their underlying local subproblems in response to queries from the coordinator. Proximal minimization is well-suited for networks with privacy constraints since it safeguards each agent's local objective and constraints from being revealed to the coordinator or other agents. Once the coordinator receives the local proximal minimization solutions from the agents, it employs them to formulate new queries for the agents, thus guiding the agents' solutions towards the global solution. These distributed optimization schemes find applications in various domains, such as smart building power management sensor networks, smart buildings, and smart manufacturing, as evidenced by [3].

Numerous algorithms are suitable for addressing distributed convex optimization; for instance, [2], [4], [5], and [6] offer relevant insights. Among these algorithms, a particularly notable method is the Alternating Direction Method of Multipliers (ADMM), initially introduced in [7]. This approach effectively tackles optimization problems by breaking them down into smaller local sub-problems. Subsequently, each agent tackles its local sub-problem and transmits its outcomes to a coordinator, which aggregates all the agents' solutions to construct the global objective. ADMM offers two key advantages: it is relatively straightforward to implement, and due to its decomposing nature, it lends itself well to parallelization. As outlined in [8], ADMM finds extensive applications in statistical and machine learning problems, including Lasso, sparse logistic regression, basis pursuit,

support vector machines, and various others. Furthermore, ADMM has been widely employed in machine learning problems, as well as in other distributed optimization scenarios [9–13].

The inherent query-response mechanism in distributed optimization algorithms, including ADMM, frequently necessitates numerous iterations before converging to a solution. However, a significant volume of communication between the coordinator and agents may render the system impractical, particularly in scenarios where communication is costly, such as underwater communication for robot formation control [14]. Therefore, minimizing communication costs is highly desirable, even crucial, for ensuring the feasibility of these distributed optimization schemes in real-world applications.

Efforts to reduce communication in distributed optimization settings have been previously explored. For example, in [15], the authors introduced a hierarchical distributed optimization algorithm tailored for predictive control in smart grids. This algorithm mitigates communication overhead by circumventing direct communication between agents, instead requiring agents to communicate solely with the coordinator at each iteration. Efficient solutions for large-scale machine learning applications leveraging distributed optimization schemes with a focus on communication efficiency have been proposed in [16, 17]. The authors of [18] successfully reduced communication complexity by employing ADMM to solve each subsystem and applying the k-means algorithm to partition a distributed smart grid. In [19], ADMM-based communication-efficient federated learning algorithms are proposed, which perform aggregation at a central coordinator of the updates sent by other agents at predefined intervals. In [20], the authors propose employing a communication censoring strategy to devise a communication-efficient ADMM algorithm for resolving

a convex consensus optimization problem. In [21], the concept of *the Moreau envelope function* is utilized, and it is further elaborated in [22], to predict the proximal operators of the local agents to facilitate skipping certain communication rounds. Similarly, in [23], the same concept is employed, where the local proximal operators and their gradients are predicted using Gaussian Processes (GP). The GP models generate estimations of prediction uncertainty, which are utilized by the coordinator to determine the necessity of communication with each agent.

Reducing the communication load can be achieved not only by directly limiting the number of communication rounds but also by addressing the total communication overhead, which includes the payload size of the information transmitted in each iteration of a distributed optimization algorithm. Payload size reduction can be accomplished by quantizing the data exchanged between agents and the coordinator. Various studies have proposed quantization methods aimed at reducing the data exchange size in each algorithmic iteration, consequently minimizing overall communication overhead. In [24], a quantized distributed composite optimization problem over relay-assisted networks was addressed using a simplified augmented Lagrangian method. In [25], a distributed optimization problem affected by quantization was tackled employing the inexact proximal gradient method. Additionally, in [26], a distributed optimization problem was resolved utilizing a distributed gradient algorithm with adaptive quantization.

The work in this dissertation aims to extend the work in [23] by adding quantization to a distributed optimization problem solved with ADMM where each agent's response is predicted by GP. Related to GP regression with quantized data is GP regression where part of the data was censored, which has been previously studied. The authors of

[27] described a GP framework in which all data outside of a specific range were fixed to a value. Furthermore, in [28], a system identification approach with quantized output data modeled with GP was presented, where Gibbs sampler was utilized for kernel hyperparameters estimation. In addition, in [29], GP was employed to predict the best locations for sensors in a spatial environment.

In our preliminary published work [1], we proposed a solution to a distributed optimization problem using ADMM, where GP regression was employed to predict the proximal operators, and the communications from agents to coordinator were quantized. However, this approach had two limitations: 1) It did not consider the quantization of the training data in optimizing the GP hyperparameters and in GP regression; and 2) It did not address the correlation between quantization noise and inputs, nor did it mitigate these correlation issues. Since GP regression assumes a joint Gaussian distribution among evaluations of the latent function, adapting the regression modeling to account for non-Gaussian quantization noise and its correlation with the original function values is essential. Failure to address this discrepancy can lead to inaccuracies in the inferred values, which in turn impacts the accuracy of the ADMM algorithm. This discrepancy may result in an increased number of iterations required to achieve convergence or even potential failure to converge altogether. Therefore, adjusting the regression modeling to better align with the characteristics of the quantization noise is crucial for the overall effectiveness of the distributed optimization process. In Chapter 2, we address these limitations by integrating two components: an adaptive uniform quantizer with *dithering* [30–32] and *joint dithering and orthogonal transformation* [33], and an improved regression method that takes into account the quantization error in the learning data. As a result, the regression

algorithm has to be revised accordingly by taking into consideration the resulting statistics of measurements in the presence of quantization noise.

In addition, in Chapter 3, we present a convergence analysis for our hybrid approach. This analysis first presents a convergence proof that relies on a query decision using the trace of the GP covariance matrix and an infinitely large quantization resolution allowed. This proof can not be used directly to prove the convergence of our proposed hybrid approach; however, it is used to show the convergence properties of our method and to demonstrate that the expectation of the ADMM residual is bounded and such bound decreases at each iteration.

We continue our study by proposing to explore how the coordinator's decision on which agents are required to communicate affects the overall performance of our communication reduction approach. In [23], the coordinator decided whether a communication with an agent is required when the maximum variance, given by the agent's corresponding GP regression, is below a certain threshold. This communication decision will be referred to as an independent query strategy, since the coordinator makes its decision using the uncertainty measure of each agent without considering the others. We propose to test different independent query strategies in addition to the one presented in [23]. Furthermore, we propose studying the inherent coupling of agents in the ADMM algorithm to develop a joint query strategy that will use a joint uncertainty measurement to decide which agents are required to communicate.

Finally, we finish this study in Chapter 5 by proposing a refined hybrid approach in which we not only account for the quantization error in the regression method but allow for a fully adaptive uniform quantization scheme. This adaptation not only adapts the

quantizer's mid-value and window length, but also assigns different quantization resolutions to each agent. The rationale of this adaptation is that not every agent contributes uniformly to the total uncertainty so, depending on the value of each agent's trace of its covariance matrix and a decaying threshold, we determine each agent quantization resolution to control the system's overall uncertainty while minimizing the overall transmission load. Finally, since this refined hybrid approach uses the trace of the regression's covariance matrix to make the communication decision, it is aligned with our derived convergence proof.

Our main contributions are summarized below.

Main Contributions:

- We study the statistics of the quantization error of the adaptive uniform quantizer proposed in our previous work [1], and characterize its impact on the distributed optimization algorithm.

- We improve the hybrid communication reduction approach in [23], which combines proximal operator learning and adaptive quantization, employing a novel Linear Minimum Mean Square Estimator (LMMSE)-based regression that takes into account the quantization error statistics. We also develop an additional LMMSE to approximate more accurately the gradient of the Moreau Envelope used in the ADMM algorithm.

- The impact of quantization error is mitigated in our learning algorithm by integrating our adaptive uniform quantizer with orthogonal transformations and dithering.

- A convergence analysis of our hybrid approach is presented. This analysis is based on a derived convergence proof that is closely tied to the trace of the covariance matrix given by the regression process and relies on an unrestricted assignment of quantization bits.

- We propose three different independent query strategies for the communication reduction approach in [23], where the coordinator solely uses the uncertainty of the prediction of each agent to decide whether such agent should be queried.

- We study the ADMM expression for the sharing problem and present a rearrangement of such expression showing the inherent coupling between agents when run-

ning ADMM.

- We propose a joint query strategy that takes into account the inherent coupling between agents, and using a joint uncertainty measurement decides which agents should be queried considering the dynamics of all agents as a whole.

- A refined hybrid approach is presented that makes its communication decision relying on the trace of the predictor's covariance matrix and considers a uniform quantizer that not only adapts its mid-value and window length but also adapts the quantization resolution according to each agent's needs.

- We validate our approach and algorithms in an extensive empirical study of a sharing problem with quadratic cost function. We present numerical experiments for a network of 10, 20, and 30 agents for which we ran 100 experiments for each. The numerical results show significant reductions in total communication expenditure in all test cases, with negligible compromise in the optimization performances.

The organization of the dissertation is given below. In Chapter 2 the hybrid approach that combines the proposed modified regression process with uniform quantization. Chapter 3 presents a convergence analysis of the ADMM algorithm to address the sharing problem when applied in conjunction with the stochastic STEP-GP algorithm [23] and its variant named LGP derived in Chapter 2. In Chapter 4 we study the ADMM expression for the sharing problem and propose different query strategies to improve the communication decision-making of our query-response approach. This chapter does not consider quantization. Then, in Chapter 5 the LGP algorithm in Chapter 2 is extended to include an adaptive quantization scheme that also adapts its quantization resolution. Finally, the global conclusions and proposed future directions are presented in Chapter 6.

Chapter 2. Hybrid Approach Combining the Modified Gaussian Process LGP and Adaptive Uniform Quantization

This chapter focuses on our proposed hybrid approach that combines adaptive uniform quantization and GP regression. The work done did not simply put these two concepts in a distributed optimization setting in a naive way, but considered the quantization error statistics to be accounted for in the regression process mitigating its impact on the ADMM algorithm.

Chapter Organization: The problem formulation is given in Section 2.1. An overview of uniform quantization and GP regression is presented in Section 2.2. Then, Section 2.3 presents the main mathematical foundation and derivations relevant to our work. A detailed presentation of our proposed approach is shown in Section 2.4. The numerical results are presented in Section 2.5. Finally, we conclude the chapter with the main contributions in Section 2.6.

2.1. Problem Formulation

This chapter deals with a multi-agent optimization problem whose structure takes the form of the sharing problem as considered in [8, 10]:

$$\text{minimize} \quad \sum_{i=1}^{n} f_i\left(x_i\right) + h\left(\sum_{i=1}^{n} x_i\right). \tag{2.1}$$

Here, n agents, each with local decision variables $x_i \in \mathbb{R}^p$, equipped with a proper and strongly convex local cost function f_i: $\mathbb{R}^p \mapsto \mathbb{R}$, coordinate to minimize the system cost consisting of all local costs and a proper and convex shared global cost function h: $\mathbb{R}^p \mapsto \mathbb{R}$. Each cost function is only known to its corresponding agent and cannot be shared with the coordinator or other agents for privacy reasons. The problem presented in (2.1) can be solved with ADMM. By introducing copies y_i of x_i, the problem can be formulated

equivalently as

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n} f_i(x_i) + h\left(\sum_{i=1}^{n} y_i\right) \\ \text{subject to} \quad & x_i - y_i = 0, \quad \forall i = 1, \ldots, n. \end{aligned} \tag{2.2}$$

Because the agents keep their local cost function f_i private, each agent i will only provide the solution to the following local *proximal minimization problem* to the coordinator

$$\mathbf{prox}_{\frac{1}{\rho} f_i}(z_i^k) = \underset{x_i \in \mathbb{R}^p}{\arg\min} \left\{ f_i(x_i) + \frac{\rho}{2} \|x_i - z_i^k\|^2 \right\}, \tag{2.3}$$

in response to a value (a query) z_i^k sent to it by the coordinator at iteration k, where $\rho > 0$ is a penalty parameter. The ADMM works in a query-response manner as follows. At iteration k, a query point z_i^k is generated by the coordinator and sent to an agent i. Each agent solves its proximal minimization problem at its query point z_i^k and replies with the response vector $\mathbf{prox}_{\frac{1}{\rho} f_i}(z_i^k)$ to the coordinator. The coordinator then updates the dual variables and generates the query points at the next iteration. Mathematically, each ADMM iteration k involves the following updates derived in the analysis in Chapter 7 in [8]:

1. The coordinator updates the average of y_i

   $$\bar{y}^{k+1} = \underset{\bar{y} \in \mathbb{R}^p}{\arg\min} \left\{ h(n\bar{y}) + (n\rho/2) \|\bar{y} - \bar{x}^k - u^k\|^2 \right\}$$

   then sends a query $z_i^k = x_i^k - \bar{x}^k + \bar{y}^{k+1} - u^k$ to each agent i.

2. Each agent i updates and sends its response $x_i^{k+1} = \mathbf{prox}_{\frac{1}{\rho} f_i}\left(z_i^k\right)$ to the coordinator.

3. The coordinator calculates the average $\bar{x}^{k+1} = (1/n) \sum_{i=1}^{n} x_i^{k+1}$ and updates the scaled dual vector $u^{k+1} = u^k + \bar{x}^{k+1} - \bar{y}^{k+1}$.

This process is repeated until convergence is achieved or until a maximum number of iterations is reached. The most common termination criterion for ADMM is presented in Section 3.3.1 in [8].

2.1.1. Moreau Envelope

To reduce the communication overhead in this distributed optimization scheme, the authors of [22] proposed an approach called STEP (STructural Estimation of Proximal operator) which relies on the concept of the Moreau envelope of a function f. For brevity, we drop the subscript i and the superscript k in the subsequent equations. For $1/\rho > 0$, the Moreau envelope $f^{\frac{1}{\rho}}$ of f is defined as

$$f^{\frac{1}{\rho}}(z) = \min_{x \in \mathbb{R}^n} \left\{ f(x) + \frac{\rho}{2}\|x - z\|^2 \right\}. \tag{2.4}$$

When f is a proper and convex function, the Moreau envelope $f^{\frac{1}{\rho}}$ is *convex and differentiable with Lipschitz continuous gradient with constant* ρ [Fact 2.2 in [34]]. Moreover, the unique solution to the proximal minimization $\mathbf{prox}_{\frac{1}{\rho}f}(z)$ is [35, Proposition 5.1.7]

$$\mathbf{prox}_{\frac{1}{\rho}f}(z) = z - \frac{1}{\rho}\nabla f^{\frac{1}{\rho}}(z). \tag{2.5}$$

Consequently, the gradient $\nabla f^{\frac{1}{\rho}}(z)$ is all that is required to reconstruct the optimizer of (2.3) following from (2.5).

The STEP approach estimates the unknown gradient $\nabla f^{\frac{1}{\rho}}(z)$ at any query point z by constructing a set of possible gradients at z based on past queries and then selecting a gradient that is "most likely" the true gradient. The work presented in [23] improved STEP by learning the Moreau envelopes corresponding to the local proximal operators with GP, which are updated online from past query data and used to predict the gradient $\nabla f^{\frac{1}{\rho}}(z)$ for estimating the proximal operators (2.3) of the agents by (2.5).

2.1.2. Proposed Solution Overview

The communication expenditure can be reduced further if the learning component is combined with the quantization of the communications between agents and coordinator.

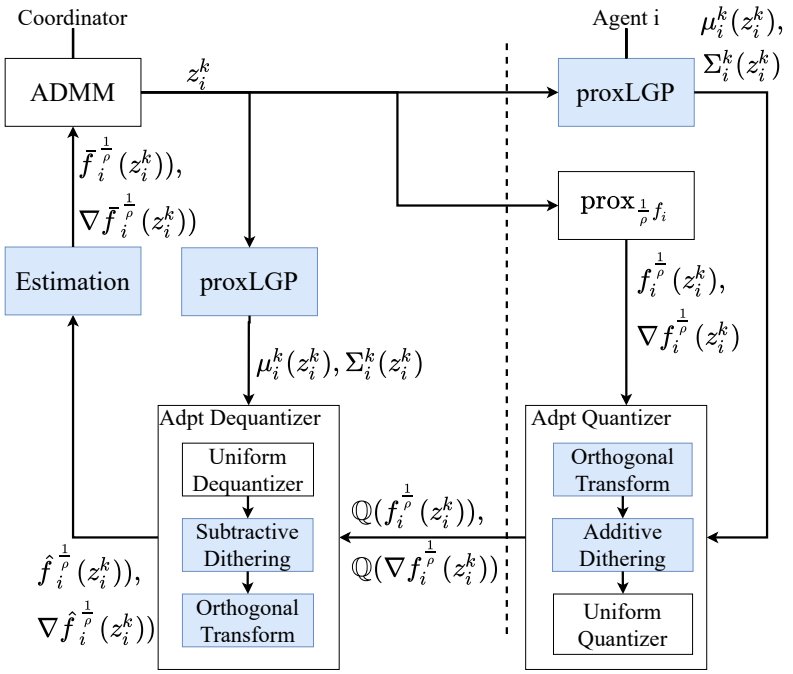

Figure 2.1. Flow diagram of a query and response between the coordinator and an agent in the proposed approach. The enhancements contributed by this work, compared with the original approach in [1], are highlighted in the blue-shaded boxes.

Our work [1] presented some preliminary results on a hybrid approach combining learning with quantization for further reducing communication overhead. This chapter builds upon our hybrid approach [1] by further analyzing and mitigating the impact of quantization errors. Our improved hybrid approach is depicted in the diagram in Figure 2.1, which describes the communication and computation processes between the coordinator and an agent i at ADMM iteration k. In the colored boxes are new or modified components developed in this work compared to the approach in [1]. The blocks colored blue indicate the processes that were added or improved compared with our work in [1].

In Figure 2.1, if the coordinator determines that a communication with agent i is necessary at iteration k, it will send the query point z_i^k to the agent. The Moreau envelope $f_i^{1/\rho}(z_i^k)$ and its gradient $\nabla f_i^{1/\rho}(z_i^k)$ are then calculated. A regression is performed

simultaneously by the agent's proxLGP (identical to the coordinator's proxLGP), to obtain the predictive mean $\mu_i^k(z_i^k)$ and the covariance matrix $\Sigma_i^k(z_i^k)$ of the agent's response. These values are used to parameterize the quantization process of the exact response $\{f_i^{1/\rho}(z_i^k), \nabla f_i^{1/\rho}(z_i^k)\}$ to reduce the quantization error. The rationale is that if the exact values fall with high probability inside a range (determined by the predictive covariance matrix) around the predictive mean, then the quantization error is reduced and diminished as the proxGP becomes increasingly accurate, ensuring the optimization's convergence [25]. The quantized response $\left\{\left(\mathbb{Q}\big(f_i^{1/\rho}(z_i^k)\big), \mathbb{Q}\big(\nabla f_i^{1/\rho}(z_i^k)\big)\right)\right\}$ from agent i is sent back to the coordinator, which uses a similar dequantization process based on the same predictive mean $\mu_i^k(z_i^k)$ and covariance matrix $\Sigma_i^k(z_i^k)$ to obtain the dequantized approximate response $\{\hat{f}_i^{1/\rho}(z_i^k), \nabla \hat{f}_i^{1/\rho}(z_i^k)\}$. The dequantized values are used both for the ADMM calculations and for updating the proxGP.

In the next section, we present a review of the important theoretical results relevant to our work.

2.2. Review of Gaussian Process and Quantization

2.2.1. Gaussian Process with Derivative Observations

Let us assume that we have m observations of a random variable, and $X \in \mathbb{R}^{m\times p}$ whose rows x_i ($i \in [1, m]$) are observed inputs vectors. Considering a mean function $\mu(x_i)$ and the co-variance function $\phi(x_i, x_i')$ of a real process $f(x_i) \in \mathbb{R}$ satisfying positive definite conditions as presented in Chapter 4 of [36], the GP can be written as $f(x_i) \sim \mathcal{GP}(\mu(x_i), \phi(x_i, x_i'))$.

Now, consider the case where we have extended function values at $x_i \in \mathbb{R}^{1\times p}$ in-

cluding both the function value and its gradients at x_i, denoted by $[f(x_i); \nabla f(x_i)]$, where $\nabla f(x_i) = \left[\frac{\partial f(x_i)}{\partial x_i^{(d)}}\right]_{d=1,\ldots,p}$, and $x_i^{(d)}$ is the d-th element of x_i. Following [37], the covariance matrix is correspondingly expanded, for any pair of points $s, l \in [1, m]$, resulting in the covariances between the observations and its partial derivatives given by

$$\text{Cov}\left[\frac{\partial f(x_s)}{\partial x_s^{(d_s)}}, f(x_l)\right] = \frac{\partial}{\partial x_s^{(d_s)}}\phi\left(x_s, x_l\right),$$

and between the partial derivatives given by

$$\text{Cov}\left[\frac{\partial f(x_s)}{\partial x_s^{(d_s)}}, \frac{\partial f(x_l)}{\partial x_l^{(d_l)}}\right] = \frac{\partial^2}{\partial x_s^{(d_s)} \partial x_l^{(d_l)}}\phi\left(x_s, x_l\right),$$

where $1 \leq d_s, d_l \leq p$. The GP then will have its predicted mean and covariance as presented in Chapter 2 of [36].

2.2.2. Uniform Quantization

We consider a uniform quantizer $\mathbb{Q}_{\text{u}}$ of the mid-tread type [38], where the input-output relation is given by

$$\mathbb{Q}_{\text{u}}(y; \bar{y}, q) = \bar{y} + q\left(\left\lfloor \frac{y - \bar{y}}{q} \right\rfloor + \frac{1}{2}\right),$$

in which $q > 0$ is the quantization window length, $\bar{y}$ is the mid-value, and $\lfloor y \rfloor$ denotes the integer closest to y towards 0. Here, $q = \frac{l}{2^b}$, where l is the range of the quantization interval and b is the bit resolution of the quantizer. Let $\hat{y} = \mathbb{Q}_{\text{u}}(y; \bar{y}, q)$, then the quantization error (or quantization noise) is defined as $\epsilon_{\mathbb{Q}} = y - \hat{y}$. The statistics of the quantization error for this uniform quantizer are characterized in Section V-A in [39].

2.3. GP Regression under Adaptive Quantization

In this section, we present the derivations and principles of our proposed approach. We present our proposed adaptive quantization scheme and its properties, the new regres-

sion mechanism, and an approximation method to deal with the quantized data.

2.3.1. Adaptive Uniform Quantization

We propose a quantizer that adapts the standard (non-adaptive) uniform quantizer. Given an input y which is a sample of a Gaussian distribution $\mathcal{N}\left(\mu_y, \sigma_y^2\right)$, we adapt a uniform quantizer by setting its mid-value $\bar{y} = \mu_y$ and its range $l = 2c\sigma_y$, for some given $c > 0$ controlling how many standard deviations apart from the μ_y we set the range which influences how confident we are that the quantizer's input is within the defined range. The proposed adaptive quantizer $\mathbb{Q}_{\text{ua}}$ on y, given by $\mathbb{Q}_{\text{ua}}(y; \mu_y, \sigma_y, c, b) = \mathbb{Q}_{\text{u}}(y; \mu_y, \frac{2c\sigma_y}{2^b}) = \mu_y + \frac{2c\sigma_y}{2^b}\left(\left\lfloor \frac{2^b(y-\mu_y)}{2c\sigma_y} \right\rfloor + \frac{1}{2}\right)$, therefore has parameters that are adapted for a quantization resolution appropriate for the most likely values of $f(x)$.

The following result characterizes the error statistics of the adaptive uniform quantizer, which will play an important role in the analysis of our proposed adaptive quantization methods throughout the rest of the chapter. Its proof is presented in Appendix A.

Proposition 1 *Consider a sample y of a Gaussian distribution $\mathcal{N}\left(\mu_y, \sigma_y^2\right)$ and an adaptive uniform quantizer $\mathbb{Q}_{\text{ua}}(y; \mu_y, \sigma_y, c, b)$ on y. Define the quantization error $\epsilon_{\mathbb{Q}} = y - \mathbb{Q}_{\text{ua}}(y; \mu_y, \sigma_y, c, b)$. Then the mean and variance of the quantization error are*

$$\mathbb{E}[\epsilon_{\mathbb{Q}}] = 0$$

$$\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}] = \frac{q^2}{12}v(r),$$

where $q = \frac{2c\sigma_y}{2^b}$, $r = \frac{2^b}{2c}$, and

$$v(r) = 1 + \frac{12}{\pi^2}\sum_{m=1}^{\infty}\frac{(-1)^m}{m^2}\exp\left(-2\pi^2 m^2 r^2\right). \tag{2.6}$$

Furthermore, the correlation between the input y and the quantization error is given by,

$$\mathbb{E}[y\epsilon_{\mathbb{Q}}] = 2\sigma_y \sum_{m=1}^{\infty} (-1)^m \exp\left(-2\pi^2 m^2 r^2\right). \tag{2.7}$$

While $v(r)$ and $\mathbb{E}[y\epsilon_{\mathbb{Q}}]$, given in (2.6) and (2.7), involve complex mathematical series, we will show that when the ratio $r = \frac{2^b}{2c}$ exceeds 1, $v(r)$ becomes approximately 1 and the correlation $\mathbb{E}[y\epsilon_{\mathbb{Q}}]$ becomes negligible. The following lemmas establish the monotonicity and the negative values of these series. Their proofs can be found in Appendix B.

Lemma 1 *The series $\sum_{m=1}^{\infty} \frac{(-1)^m}{m^2} \exp\left(-2\pi^2 m^2 r^2\right)$ is negative and increasing with r.*

Lemma 2 *The series $\sum_{m=1}^{\infty} (-1)^m \exp\left(-2\pi^2 m^2 r^2\right)$ is negative. Furthermore, for $r > \frac{1}{\sqrt{2\pi}} \approx 0.225$, it is increasing with r.*

It follows from these lemmas that $v(r) < 1$ and increasing with r for all $r > 0$, and $\mathbb{E}[x\epsilon_{\mathbb{Q}}] < 0$ and increasing with r for all $r > \frac{1}{\sqrt{2\pi}} \approx 0.225$. In practice, the ratio $r = \frac{2^b}{2c}$ is at least 1 and often much greater than 1. Indeed, with the typically chosen $c = 3$, at a resolution of just $b = 3$ bits, $r = 4/3 > 1$ and increases exponentially with b. At $r = 1$, we have $v(1) = 1 - 3.253 \times 10^{-9}$, and $\mathbb{E}[y\epsilon_{\mathbb{Q}}] = -5.351 \times 10^{-9}\sigma_y$. Therefore, for all practical purposes, we have $1 - 3.253 \times 10^{-9} \leq v(r) < 1$, thus we can consider $v(r) = 1$ and hence $\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}] = \frac{q^2}{12}$. In addition, we have $-5.351 \times 10^{-9}\sigma_y \leq \mathbb{E}[y\epsilon_{\mathbb{Q}}] < 0$, thus we can consider $\mathbb{E}[y\epsilon_{\mathbb{Q}}] = 0$.

2.3.2. Adaptive Uniform Quantization with Vector Input

Consider the case where the input to the quantizer is a Gaussian random vector y with conditional mean vector μ_y and conditional co-variance matrix Σ_y. The previously presented adaptive quantization scheme must be adjusted to handle the multidimensional nature of the input. We propose two schemes described below: one ignores the correlations

among the input values and the other takes these correlations into account.

2.3.2.1. Adaptive Scheme Ignoring Correlation

Quantization is performed element-wise, using each element of the quantizer's input with its corresponding element of the conditional mean vector μ_y and the diagonal of the co-variance matrix Σ_y for adaptation. Therefore, we have a vector of window lengths q with the i^{th} entry given by

$$q_i = \frac{2c\sqrt{\Sigma_{y[ii]}}}{2^b}, \tag{2.8}$$

where $\Sigma_{y[ii]}$ is the i^{th} entry of the diagonal of Σ_y. Using Proposition 1, we can characterize the quantization error under the proposed scheme, as stated in the following proposition.

Proposition 2 *Under the Adaptive Scheme Ignoring Correlation, an adaptive uniform quantizer* $\mathbb{Q}_{\text{ua}}(y; \mu_y, \Sigma_y, c, b)$ *has a quantization error vector* $\epsilon_{\mathbb{Q}}$ *whose components are uncorrelated. The correlation matrix, defined as* $\Delta_{\text{un}} = \mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]$*, is a diagonal matrix with its diagonal given by the vector* $\frac{v(2^b/2c)}{12}q^2$*, with the entries of vector* q *defined as in* (2.8) *and* $v(\cdot)$ *as defined in Proposition 1.*

2.3.2.2. Correlated Adaptive Scheme

The use of an orthogonal transformation of the quantizer's input y allows us to consider the correlation between its elements, and to perform quantization over the transformed input similarly as in the previously defined *Adaptive Scheme Ignoring Correlation*.

Using the above notations, the orthogonal transformation to the quantizer's input is expressed as

$$y^A = A(y - \mu_y), \tag{2.9}$$

where A is the transformation matrix. The conditional mean of y is subtracted to have

a zero-mean quantizer's input. Then, the way A is determined will define our orthogonal *pre-filtering* of the quantizer's input.

Pre-filtering: *The transformation matrix A used in* (2.9) *is obtained by applying an eigenvalue decomposition of matrix Σ_y, in which $\Sigma_y = U\Lambda U'$, with Λ being a diagonal matrix with the eigenvalues of Σ_y and U being a square matrix whose columns are eigenvectors of Σ_y. The matrix A can be expressed in two ways; $A_1 = (\Sigma_y)^{-1/2}$ or $A_2 = U'$, where $(\Sigma_y)^{1/2}$ is a matrix such that $(\Sigma_y)^{1/2}(\Sigma_y)^{1/2} = \Sigma_y$. The use of A_1 will result in a whitening procedure where the result will be a zero-mean unit variance vector with independent components. The use of A_2 will result in a decoupling procedure where the result will be a zero-mean vector whose variances are determined by the eigenvalues in Λ.*

Following this pre-filtering, y^A will be element-wise quantized given by:

$$\mathbb{Q}_{\text{ua}}(y^A; 0, \Sigma_w, c, b) = y^A + \epsilon_{\mathbb{Q}},$$

where Σ_w represents the identity matrix (when $A = A_1$) or a diagonal matrix with entries given by the eigenvalues of Σ_y (when $A = A_2$).

Proposition 3 *Under the Correlated Quantization Scheme and the proposed Pre-filtering, an adaptive uniform quantizer $\mathbb{Q}_{\text{ua}}(y^A; 0, \Sigma_y, c, b)$, where the input vector is transformed following* (2.9)*, has a quantization error vector $\epsilon_{\mathbb{Q}}$ whose components are correlated with each other. The correlation matrix, defined as $\Delta\text{co} = E[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]$, is independent of the choice of the transformation matrix A and is given by $\Delta\text{co} = \frac{c^2 v(2^b/2c)}{3(2^b)^2}\Sigma_y$, with $v(\cdot)$ as defined in Proposition 1.*

Proof: The proof is presented in Appendix C. □

2.3.3. LMMSE Regression with Quantization

In this subsection, we consider a GP regression as presented in Section 2.2.1, but when the training set $\mathcal{D}$ is affected by adaptive quantization. In this scenario, we do not have access to the exact extended values y_i but a quantized version of them $\hat{y}_i = [\mathbb{Q}_{\mathrm{u}}(f(x_i)); \mathbb{Q}_{\mathrm{u}}(\nabla f(x_i))^T] + \epsilon_{\mathrm{n}}^i$, which are quantized following the proposed adaptive quantization with vector inputs presented in Section 2.3.2. These quantized extended values are also expressed as $\hat{y}_i = [f(x_i); \nabla f(x_i)^T] + \epsilon_{\mathrm{n}}^i + \epsilon_{\mathbb{Q}}^i$, where $\epsilon_{\mathbb{Q}}^i$ refers to the quantization error vector for the observation i and ϵ_{n}^i is a vector whose entries follow the same Gaussian distribution with zero mean, σ_n^2 variance at observation i. Such Gaussian noise is not a physical noise but one added to avoid possible matrix singularity.

The added non-Gaussian quantization noise invalidates the Gaussian noise assumption of the regular GP regression. In this case, the regression cannot be a Minimum Mean Square Estimator (MMSE) anymore, so we must compute the conditional mean which requires a more involved computation. To overcome this challenge, we adopt a Linear Minimum Mean Square Error Estimator (LMMSE). This allows us to balance the accuracy and complexity of the estimator while preserving the advantages of GP. With this premise we will derive two estimators under two scenarios regarding the training set $\mathcal{D}$.

2.3.3.1. Linear GP Regression (LGP-R)

This estimator is used to predict the extended values of an input x_* given a training set where the observed extended values are affected by quantization. In this case, we only have access to quantized values of the extended values. For a new input x_* we want to predict y_*, leading to the following theorem, whose proof is presented in Appendix D.

This estimation is done at every iteration, and for every agent to assess the quality of regression.

Theorem 1 *The LGP-R Estimator has an input $x_* \in \mathbb{R}^p$ and a training set containing m past observations with quantized extended values $\mathcal{D} = (X, \hat{Y})$, with $X \in \mathbb{R}^{m(p+1)\times p}$ being a collection of the past input observations $x_i \in \mathbb{R}^{(p+1)\times p}$, and $\hat{Y} \in \mathbb{R}^{m(p+1)\times 1}$ being a collection of the past quantized extended values $\hat{y}_i \in \mathbb{R}^{(p+1)\times 1}$. This estimator has its predicted mean*

$$\mu(x_*) = \Phi(X_*, X)(\Phi(X, X) + \sigma_n^2 I_{m(p+1)} + \Delta + 2\mathbb{E}[Y\epsilon'_{\mathbb{Q}}])^{-1}\hat{Y},$$

and predicted covariance matrix

$$\Sigma(x_*) = \Phi(X_*, X_*) - \Phi(X_*, X)(\Phi(X, X) + \sigma_n^2 I_{m(p+1)} + \Delta + 2\mathbb{E}[Y\epsilon'_{\mathbb{Q}}])^{-1}\Phi(X, X_*),$$

where $X_* \in \mathbb{R}^{(p+1)\times p}$ contains a copy of x_* in each of its rows, the entries of the matrices $\Phi(X_*, X_*)$, $\Phi(X_*, X)$, and $\Phi(X, X)$ are as detailed in Subsection 2.2.1, $\Delta = \mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]$ contains the information of the uniform quantization error of all extended values observations of the training set $\mathcal{D}$, and the entries corresponding to each observation in Δ are added block-wise following the expression given by Δ_{un} in Proposition 2 or Δ_{co} in Proposition 3 (depending on the quantization scheme selected), and $\mathbb{E}[Y\epsilon'_{\mathbb{Q}}]$ is the correlation between the uniform quantization error of all extended values observations of the training set $\mathcal{D}$ and the extended values observations, whose entries are calculated following the correlation expression shown in Proposition 1.

2.3.3.2. Linear GP Approximation (LGP-A)

Consider the case where we perform adaptive uniform quantization on the extended values at x_*, resulting in the quantized version of y_* given by $\hat{y}_*$. Such adaptive quanti-

zation uses the conditional mean and conditional covariance given by LGP-R. It is possible to approximate the real value y_* if $\hat{y}_*$ and the statistics that adapt the quantizer are known. To do so, we propose the construction of a LMMSE named LGP-A to be performed after the quantization process. This estimation is only performed when communication is required and after receiving the reply from the agent.

The estimation could be performed by updating the training set with the new input and the quantized extended values. Input x_* could then be reinserted to the estimator presented in Theorem 1. To avoid such redundancy we consider an approximator that deals with a zero-mean input $\hat{y}_* - \mu(x_*)$, and since $\hat{y}_*$ already has the information of the past training set, we then have the following theorem, whose proof is presented in Appendix E.

Theorem 2 *LGP-A has a training set containing past observations and extended quantized values of x_* leading to the set $\mathcal{D} = ([X; x_*], [\hat{Y}; \hat{y}_*])$, with $X \in \mathbb{R}^{m(p+1)\times p}$ being a collection of the past input observations $x_i \in \mathbb{R}^{(p+1)\times p}$, and $\hat{Y} \in \mathbb{R}^{m(p+1)\times 1}$ being a collection of the past quantized extended values $\hat{y}_i \in \mathbb{R}^{(p+1)\times 1}$. LGP-A estimates the target value y_* by*

$$\bar{y}_* = B(\hat{y}_* - \mu(x_*)) + \mu(x_*),$$

where $B = \Sigma(x_*)(\Sigma(x_*) + \Delta_{p+1} + \sigma_n I_{p+1} + 2\mathbb{E}[y_* \epsilon'_{\mathbb{Q}*}])^{-1}$, with $\mu(x_*)$ and $\Sigma(x_*)$ as presented in Theorem 1 and Δ_{p+1} is given by Δ_{un} in Proposition 2 or Δ_{co} in Proposition 3 depending on the quantization scheme selected, $\epsilon_{\mathbb{Q}*}$ is the quantization error of only the quantized values in the present iteration, and $\mathbb{E}[y_* \epsilon'_{\mathbb{Q}*}]$ is calculated as shown in Proposition 1.

2.4. Proposed Approach

2.4.1. Proposed Adaptive Uniform Quantization Scheme

This section combines the overview presented in Section 2.1 with the results presented in Section 2.3 to present our complete proposed approach in more detail.

In Figure 2.1, upon receiving the query point $z_i^k \in \mathbb{R}^{1\times p}$ from the coordinator (left side), agent i (right side) solves the proximal minimization problem (2.3) (the box $\mathbf{prox}_{1/\rho f_i}$) and obtains the exact values of $f_i^{1/\rho}(z_i^k) \in \mathbb{R}$ and $\nabla f_i^{1/\rho}(z_i^k) \in \mathbb{R}^{p\times 1}$. Simultaneously, it uses the regression process, depicted in the block 'proxLGP', to obtain the conditional mean $\mu_i^k(z_i^k)$, which stores the predicted values of $f_i^{1/\rho}(z_i^k)$ and $\nabla f_i^{1/\rho}(z_i^k)$, and the conditional covariance matrix $\Sigma_i^k(z_i^k)$. We can adopt the same adaptive uniform quantization scheme presented in Section 2.3.1, as the exact values follow a Gaussian distribution (under the LGP model). We will denote the quantized values of the query response as $[\hat{f}_i^{1/\rho}(z_i^k); \nabla \hat{f}_i^{1/\rho}(z_i^k)] = \mathbb{Q}_{\text{ua}}([f_i^{1/\rho}(z_i^k); \nabla f_i^{1/\rho}(z_i^k)]; \mu_i^k(z_i^k), \Sigma_i^k(z_i^k), c, b))$. The output of the quantizer is transmitted from the agent (right side) to the coordinator (left side). The dequantized values $\hat{f}_i^{1/\rho}(z_i^k)$ and $\nabla \hat{f}_i^{1/\rho}(z_i^k)$ are used by the ADMM algorithm and to update the corresponding 'proxLGP' of agent i.

2.4.2. LGP-R based Regression in our Proposed Approach

The 'proxLGP' block on the coordinator side of Figure 2.1 runs at every iteration and its resulting covariance matrix is used to determine whether to send z_i^k to agent i. Using the quantization scheme for vector inputs $\mathbb{Q}_{\text{ua}}$ (defined in Section 2.3.2) and following (2.8), the results presented in Propositions 1-3 apply to the adaptive quantizer $\mathbb{Q}_{\text{ua}}$. Hence, we can use the previously derived regression scheme LGP-R presented in Theorem

1 as the regression scheme to be used in this work. Using the results in Section 2.3.1 that $\mathbb{E}[y\epsilon'_{\mathbb{Q}}] \approx 0$ and $v(r) \approx 1$, we henceforth remove the correlation $\mathbb{E}[y\epsilon'_{\mathbb{Q}}]$ present in Theorems 1 and 2, and remove the term $v(2^b/2c)$ used in the characterization of the variance of the quantization error in Propositions 2 and 3.

Now, defining $g_i^{1/\rho}(z_i^k) = [f_i^{1/\rho}(z_i^k); \nabla f_i^{1/\rho}(z_i^k)]$, we have that, given the new query point z_i^k, the predicted value of the vector $g_i^{1/\rho}(z_i^k)$ using LGP-R will be given by

$$\mu_i^k(z_i^k) = \Phi(Z_{i*}^k, Z_i^k)(\Phi(Z_i^k, Z_i^k) + \sigma_n^2 I_{m(p+1)} + \Delta_i)^{-1}\hat{G}_i^k, \tag{2.10}$$

where $Z_{i*}^k \in \mathbb{R}^{(p+1)\times p}$ contains a copy of z_i^k in each of its rows, Z_i^k is the training input set containing queries sent to agent i up to time k in the set $\{z_i^j\}_{j\in\mathcal{J}_i}$, $\mathcal{J}_i^k$ contains the indices of the iterations where a query was sent to agent i by the coordinator up to the current algorithmic iteration, m is the number of elements in set $\mathcal{J}_i^k$, $\hat{G}_i^k$ is the quantized training target set containing the local quantized proximal minimization problem results sent from agent i to the coordinator up to time k in the set $\{\mathbb{Q}_{\text{ua}}(g_i^{1/\rho}(z_i^j); \mu_i^j(z_i^j), \Sigma_i^j(z_i^j), c, b)\}_{j\in\mathcal{J}_i}$, $\sigma_n^2 I_{m(p+1)}$, Δ_i are defined in Theorem 1, and the entries of $\Phi(Z_{i*}^k, Z_i^k)$ and $\Phi(Z_i^k, Z_i^k)$ are detailed in Subsection 2.2.1 with a covariance function given by the square exponential kernel function.

Using the same notation, the covariance matrix given by the LGP-R is

$$\Sigma_i^k(z_i^k) = \Phi(Z_{i*}^k, Z_{i*}^k) - \Phi(Z_{i*}^k, Z_i^k)(\Phi(Z_i^k, Z_i^k) + \sigma_n^2 I_{m(p+1)} + \Delta_i)^{-1}\Phi(Z_i^k, Z_{i*}^k). \tag{2.11}$$

The matrix Δ_i will be updated block-wise by inserting the corresponding quantization error covariance matrix of the query round, which follows Proposition 2 or Proposition 3 depending on the quantization scheme used. Henceforth, we will use Δ_i^k to refer to the

Algorithm 1 LGP: Distributed Optimization with Estimated Proximal Operator based on Gaussian Processes with Adaptive Uniform Quantization

Require: $x_i^0 \in \mathbb{R}^p$, $\bar{y}^0 \in \mathbb{R}^p$, $u^0 \in \mathbb{R}^p$, $c \in \mathbb{N}$, $b \in \mathbb{N}$

for $k = 0, 1, \ldots, k_{\text{stop}}$ **do**
  $\bar{y}^{k+1} \leftarrow \arg\min_{\bar{y} \in \mathbb{R}^p} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^k - u^k\|^2 \right\}$
  for each agent i **do**
    $z_i^k \leftarrow x_i^k - \bar{x}^k + \bar{y}^{k+1} - u^k$
    Calculate $\mu_i^k(z_i^k)$ and $\Sigma_i^k(z_i^k)$ from (2.10) and (2.11)
    if $\max\left(\text{diag}\left(\Sigma_i^k(z_i^k)\right)\right) > \psi_i^k$ **then**
      Send z_i^k to Agent i
      $\hat{g}_i^{1/\rho} \leftarrow \text{QUERYAGENT}(z_i^k)$ ▷ Agent i
      Compute $\bar{g}_i^{1/\rho}$ from (2.12)
      Add $(z_i^k, \hat{g}_i^{1/\rho}(z_i^k))$ to the GP training set
      Perform the GP hyperparameter update.
      $x_i^{k+1} \leftarrow z_i^k - (1/\rho)\nabla \bar{f}_i^{1/\rho}(z_i^k)$
    else
      $x_i^{k+1} \leftarrow z_i^k - (1/\rho)\mu_i^k(z_i^k)$
    end if
  end for
  $\bar{x}^{k+1} \leftarrow (1/n)\sum_{i=1}^n x_i^{k+1}$
  $u^{k+1} \leftarrow u^k + \bar{x}^{k+1} - \bar{y}^{k+1}$
  If $\|\bar{x}^k - \bar{y}^k\|_\infty \leq \epsilon_p(1 + \|\lambda^k/\rho\|_\infty)$ then Terminate.
end for

resulting quantization error covariance matrix obtained after a query process in iteration k, which will be then added to Δ_i.

2.4.3. LGP-A Approximation in our Proposed Approach

In Figure 2.1 we can see that the coordinator receives the quantized version $\nabla \hat{f}_i^{1/\rho}(z_i^k)$ of the exact value $\nabla f_i^{1/\rho}(z_i^k)$. To improve the accuracy of the gradient values used in the ADMM updates at the coordinator, we estimate these values with a LMMSE estimator rather than using the inexact quantized values directly. The estimator derived in this subsection is different from that in subsection 2.4.2 because it is applied only when a query is performed, which only uses the newly added entry in the training set. The result is further used by the ADMM process.

After a query undergoes a communication round, the quantized value of $g_i^{1/\rho}(z_i^k)$, $\hat{g}_i^{1/\rho}(z_i^k)$, is added to the regression training set, and Δ_i is updated with the block Δ_i^k. Therefore, we can obtain the desired approximation $\bar{g}_i^{1/\rho}(z_i^k)$ following the derivation from Theorem 2, which gives us

$$\bar{g}_i^{1/\rho}(z_i^k) = (B_i^k(\hat{g}_i^{1/\rho}(z_i^k) - \mu_i^k(z_i^k)))) + \mu_i^k(z_i^k), \tag{2.12}$$

where $B_i^k = \Sigma_i^k(z_i^k)(\Sigma_i^k(z_i^k) + \sigma_n I_{p+1} + \Delta_i^k)^{-1}$.

2.4.4. Dithering

From Proposition 1, we have that the correlation between the quantization noise and the input is negligible when the quantization bit resolution (b) becomes larger and we fix a small value for c. If b is too small, we can introduce dithering to randomize the quantization error and break the correlation between this error and the quantizer input.

A recent study ([40]) explores the use of quantization with dithering to determine which distribution the subtractive dithering follows. The work presented in [33] shows that the use of dithering with quantization could be improved if an orthogonal transformation was performed on the quantizer input before the quantization process. We thus adopt dithering as part of quantization after orthogonal transformation is performed at the quantizer's input.

When the uniform quantizer is used with a zero-mean Gaussian input, the dithering variable d_i^k will be a random number coming from a uniform distribution $d_{i[r]}^k \sim \mathcal{U}(\frac{-q_{i[r]}^k}{2}, \frac{q_{i[r]}^k}{2})$, where the window length $q_{i[r]}^k$ is as defined in (2.8). The dithering will be performed element-wise, so d_i^k will have the same dimension as the quantizer input. Following the orthogonal transformation as in Section 2.3.2, the quantizer input with dither-

Algorithm 2 Query Process at the Agent Side

procedure QUERYAGENT(z_i^k)
  Compute $f_i^{1/\rho}(z_i^k)$ and $\nabla f_i^{1/\rho}(z_i^k)$ from (2.4)
  $g_i^{1/\rho} \leftarrow [f_i^{1/\rho}(z_i^k); \nabla f_i^{1/\rho}(z_i^k)]$
  if Using *Adaptive Scheme Ignoring Correlation* **then**
    $\hat{g}_i^{1/\rho} \leftarrow \mathbb{Q}_{\text{ua}}(g_i^{1/\rho}; \mu_i^k(z_i^k), \Sigma_i^k(z_i^k), c, b))$
  else
    Perform decomposition $\Sigma_i^k(z_i^k) = U_i^k \Lambda_i^k U_i^{k'}$
    if Using *Whitening Transformation* **then**
      $A_i^k \leftarrow (\Sigma_i^k(z_i^k))^{-1/2}$
    end if
    if Using *Decoupling Transformation* **then**
      $A_i^k \leftarrow U_i^{k'}$
    end if
    $g_i^A \leftarrow A_i^k[g_i^{1/\rho} - \mu_i^k(z_i^k)]$
    if Using *Dithering* **then**
      Compute $g_i^{A[d]}$ as in (2.13)
      $\hat{g}_i^{1/\rho} \leftarrow \mathbb{Q}_{\text{ua}}(g_i^{A[d]}; 0, \Sigma_i^k(z_i^k), c, b)) + \mu_i^k(z_i^k)$
    else
      $\hat{g}_i^{1/\rho} \leftarrow \mathbb{Q}_{\text{ua}}(g_i^A; 0, \Sigma_i^k(z_i^k), c, b)) + \mu_i^k(z_i^k)$
    end if
  end if
  return $\hat{g}_i^{1/\rho}$
end procedure

ing is given by

$$g_i^{A[d]}(z_i^k) = g_i^A(z_i^k) + d_i^k, \tag{2.13}$$

where $g_i^A(z_i^k) = A(g_i^{1/\rho}(z_i^k) - \mu_i^k(z_i^k))$, with A as presented in the *Pre-filtering*. Then, $g_i^{A[d]}(z_i^k)$ will be quantized and sent to the coordinator. The coordinator then performs the dequantization process and subtracts the noise added to the input before adding back its mean. The value $\hat{g}_i^{1/\rho}(z_i^k)$ is given by

$$\hat{g}_i^{1/\rho}(z_i^k) = A^{-1}((g_i^{A[d]}(z_i^k) + \epsilon_{\mathbb{Q}i}^k - d_i^k) + \mu_i^k(z_i^k),$$

where $\epsilon_{\mathbb{Q}i}^k$ is the i^{th} agent quantization noise at iteration k.

2.4.5. LGP Pseudo-Code

The complete LGP algorithm considering all its different variations is presented in Algorithm 1.

2.5. Numerical Experiments

In this section, we evaluate the methods proposed in this work by solving a sharing problem where the agent's sub-problems are quadratic. The specifics of the sharing problem considered, the experiment settings, and the results obtained are presented next.

2.5.1. Sharing Problem

2.5.1.1. Problem Definition

Our testing problem is based on the application presented in [10]. In this example, a dynamic sharing problem where the problem's variables change at each iteration is presented and solved via ADMM. In our work, those varying variables are fixed and do not vary at each algorithmic step. We consider the following sharing problem:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n}(x_i-\theta_i)^T\Upsilon_i(x_i-\theta_i)+\zeta\|\sum_{i=1}^{n}y_i\|_1 \\ \text{subject to} \quad & x_i-y_i=0 \end{aligned} \tag{2.14}$$

where $x_i, y_i \in \mathbb{R}^p$, $\theta_i \in \mathbb{R}^p$, $\Upsilon_i \in \mathbb{R}^{p\times p}$ positive definite, and $\zeta > 0$ are given problem parameters.

As presented in [10], the problem in (2.14) can be applied to data flow in communication networks or currents in power grids, where there are n subsystems and p quantities distributed over such subsystems. The vector x_i describes the p quantities at subsystem i, and the goal is to determine the solution vectors x_i, $i = 1, 2, \ldots, n$.

2.5.1.2. Generation of Parameters θ_i and Υ_i

In [10] the variables θ_i and Υ_i are updated at each iteration of the ADMM algorithm. In this work, those variables are fixed by following the variable's initialization for the first iteration made in [10]. As such, to calculate each θ_i we first create θ_i^0 which is a p-dimensional vector with entries randomly generated and uniformly distributed on [-1,1]. Then, the value of θ_i to be used is $\theta_i = \theta_i^0 + \eta u_i$, where η is some small positive number, u_i is a p-dimensional vector for agent i whose entries are randomly generated and uniformly distributed on [-1,1].

Next, to calculate each Υ_i we first create $\Upsilon_i^0 = A * A'$ as a symmetric $p \times p$ matrix, where the entries of $A \in \mathbb{R}^{p \times p}$ are randomly generated and uniformly distributed on [-1,1]. Then, we generate $\widetilde{\Upsilon}_i = \Upsilon_i^0 + \eta E_i$, where E_i is a symmetric $p \times p$ matrix whose entries are randomly generated and uniformly distributed on [-1,1]. Subsequently, Υ_i is constructed as

$$\Upsilon_i = \begin{cases} \widetilde{\Upsilon}_i, & \text{if } \lambda_{min}(\widetilde{\Upsilon}_i) > \epsilon \\ \widetilde{\Upsilon}_i + \big(\epsilon - \lambda_{min}(\widetilde{\Upsilon}_i)\big) I_p, & \text{otherwise,} \end{cases}$$

where $\lambda_{min}(\widetilde{\Upsilon}_i)$ denotes the smallest eigenvalue of $\widetilde{\Upsilon}_i$ and $\epsilon > 0$ is some positive constant.

2.5.1.3. Solution with ADMM

The problem presented in (2.14) has the same form as (2.2) in Section 2.1 based on which the ADMM updates for this case are expressed as

$$\begin{aligned} x_i^{k+1} &= \underset{x_i \in \mathbb{R}^p}{\arg\min} \left\{ f_i(x_i) + (\rho/2)\|x_i - z_i^k\|_2^2 \right\} \\ \bar{y}^{k+1} &= \underset{\bar{y} \in \mathbb{R}^p}{\arg\min} \left\{ \zeta \|n\bar{y}\|_1 + (n\rho/2)\|\bar{y} - \bar{x}^{k+1} - (1/\rho)\lambda^k\|_2^2 \right\} \\ \lambda^{k+1} &= \lambda^k + \rho(\bar{x}^{k+1} - \bar{y}^{k+1}) \end{aligned} \tag{2.15}$$

Table 2.1. Elements associated with each of the proposed methods.

	GP Reg	LGP Reg	Uni Quant	Decoup	Whitening	Dithering
Sync:UniQuant			✓			
STEP-GP:Exact	✓					
STEP-LGP:UniAd		✓	✓			
STEP-LGP:UniAd-Dec		✓	✓	✓		
STEP-LGP:UniAd-DecDit		✓	✓	✓		✓
STEP-LGP:UniAd-Whit		✓	✓		✓	
STEP-LGP:UniAd-WhitDit		✓	✓		✓	✓

where $f_i(x_i) = (x_i - \theta_i)^T \Upsilon_i (x_i - \theta_i)$, $\bar{x}^k = (1/n)\sum_{i=1}^n x_i^k$, $\bar{y}^k = (1/n)\sum_{i=1}^n y_i^k$, and $z_i^k = x_i^k - \bar{x}^k + \bar{y}^k - (1/\rho)\lambda^k$.

Since the functions f_i and the l_1 norm are strongly convex, the ADMM updates for x_i^{k+1} and $\bar{y}^{k+1}$ are solutions to unconstrained convex optimization problems. Thus, those problems can be solved by calculating the derivatives of the objective functions in (2.15), and setting them equal to zero. Following this, x_i^{k+1} can be expressed by the closed form solution

$$x_i^{k+1} = (2\Upsilon_i + \rho I_p)^{-1}(2\Upsilon_i \theta_i + \rho(x_i^k - \bar{x}^k + \bar{y}^k) - \lambda^k), \tag{2.16}$$

where I_p is the $p \times p$ identity matrix.

Similarly, the $\bar{y}$ update can expressed as

$$\bar{y}^{k+1} = \begin{cases} (\bar{x}^{k+1} + \lambda^k/\rho) - \frac{\zeta}{\rho}, & \text{if } \bar{x}^{k+1} + \lambda^k/\rho > \frac{\zeta}{\rho} \\ 0, & \text{if } |\bar{x}^{k+1} + \lambda^k/\rho| \leq \frac{\zeta}{\rho} \\ (\bar{x}^{k+1} + \lambda^k/\rho) + \frac{\zeta}{\rho}, & \text{if } \bar{x}^{k+1} + \lambda^k/\rho < -\frac{\zeta}{\rho}. \end{cases} \tag{2.17}$$

2.5.2. Experiment Implementation

We consider two cases where $n \in \{10, 30\}$. The problem described in (2.14) is solved with four different methods:

1. *Direct*: this method uses a convex solver to solve the problem directly. The knowledge of the true solution is used to construct the comparative metric, which is introduced in the following subsection.

2. *Sync*: this algorithm uses ADMM with proximal operator as in (2.15), which simplifies to (2.16) and (2.17) with $\rho = 10$.

3. *STEP-GP*: the algorithm proposed in [23] combining ADMM with proximal operator with GP regression.

4. *STEP-LGP*: the hybrid algorithm proposed in this chapter, which combines the regression algorithm developed in Section 2.4.2, the LMMSE approximation presented in Section 2.4.3, and the adaptive quantization method developed in Section 2.4.1.

For each of the above algorithms, different quantization methods, or no quantization at all, are considered as follows:

- *Exact*: this method does not employ any quantization but uses 64-bit floating point numbers.

- *UniQuant*: this uniform quantization adaptation scheme is proposed in [25] to quantize the communications between agents in a connected network using the Proximal Gradient Method (PGM). In case the quantizer's input is a vector the quantization is performed element-wise. For each element of the quantizer's input, an initial quantizer's range is set which decreases at a linear rate over the algorithmic iterations and the quantizer's mid-value is set to be the previous quantized value.

- *UniAd*: this is the adaptive uniform quantization method as presented in Section 2.4.1 and performed element-wise following the *Uncorrelated Adaptive Scheme* as presented in Section 2.3.2.1.

- *UniAd-Dec*: this is the adaptive uniform quantization method as presented in Section 2.4.1 and following the *Correlated Quantization Scheme* as presented in Section 2.3.2.2 with decoupling.

- *UniAd-DecDit*: same as *UniAd-Dec* but adding the dithering procedure as presented in Section 2.4.4.

- *UniAd-Whit*: this is the adaptive uniform quantization method as presented in Section 2.4.1 and following the *Correlated Quantization Scheme* with whitening.

- *UniAd-WhitDit*: same as *UniAd-Whit* but adding the dithering procedure as pre-

sented in Section 2.4.4.

In our experiments, we consider the following combinations: *Sync:Exact*, *Sync:UniQuant*, *STEP-GP:Exact*, *STEP-LGP:UniAd*, *STEP-LGP:UniAd-Dec*, *STEP-LGP:UniAd-DecDit*, *STEP-LGP:UniAd-Whit*, and *STEP-LGP:UniAd-WhitDit*. Table 2.1 summarizes each proposed combination's algorithmic components.

The experiments were implemented in MATLAB. The solution of the minimization problems (2.14) are obtained directly using a convex solver from the YALMIP toolbox [41]. We used the GPstuff toolbox [42] for the regression training and inference. The computation was conducted with high-performance computational resources provided by Louisiana State University (http://www.hpc.lsu.edu).

2.5.3. Metrics and Considerations

2.5.3.1. MAC Metric

To consider a more realistic communication process, we include a numerical experiment component to reflect the channel contention. By modifying the simulator in [43], we get that the total transmission time will be $Tx_t = \sum_{k=1}^{N} T^k_{\text{round}}$, where N is the number of iterations taken to reach convergence, and T^k_{round} is the expected transmission time in one iteration round. Appendix F presents the specifics of how this metric was obtained.

2.5.3.2. ADMM Termination Criterion

We propose a termination criterion for ADMM using the concept of primal-residual as shown in [8], having the form:

$$\|\bar{x}^k - \bar{y}^k\|_\infty \leq \epsilon_p(1 + \|\lambda^k/\rho\|_\infty),$$

where x^k, y^k, and λ^k are the variables used in the ADMM (see Section 2.1) and ϵ_p is an adjustable tolerance whose value will affect the trade-off between communication reduction and accuracy.

2.5.3.3. Performance Metric

To compare our results, we propose the *Log Optimality over Transmission time (LOT)* performance metric

$$LOT = -\log(|J_{gt} - J_*|/J_{gt})/Tx_t$$

where J_{gt} is the true optimal value obtained by the *Direct* method, J_* is the objective value obtained by a particular approach, and Tx_t the total transmission time defined in Section 2.5.3.1. This metric reflects both communication cost and efficacy of a given approach. In particular, we want both the absolute error in the numerator and the transmission time in the denominator to be small, hence a higher LOT value is better.

2.5.3.4. Querying Mechanism

The coordinator decides if a query should be sent to agent i using a heuristic criterion utilizing the maximum component of the diagonal of the covariance matrix of the gradients of the Moreau Envelope. Specifically, if $\max\left(\text{diag}\left(\Sigma_i^k(z_i^k)\right)\right) > \psi_i^k$ then communication is needed, otherwise it is not. The threshold ψ_i^k is adapted at the coordinator side based on the setting of an initial threshold which will decrease at each iteration according to a decay rate α, such that $0 < \alpha < 1$. At k_0, which is the iteration where the GP regression is used for the first time, the initial threshold for agent i ($\psi_i^{k_0}$) is calculated following $\psi_i^{k_0} = \iota \max\left(\text{diag}\left(\Sigma_i^{k_0}(z_i^{k_0})\right)\right)$, where $0 < \iota < 1$. At iteration $k > k_0$, no matter the communication decision made by agent i, the threshold will be updated as $\psi_i^k = \psi_i^{k_0}(\alpha)^{k-k_0}$.

2.5.4. Numerical Experiments Results with $p = 5$

In this subsection, we present the results for 10 and 30 agents when the dimension of the variables is set to be $p = 5$. We also set the variable ι for the querying mechanism described in Section 2.5.3.4 to be 0.6 for all agents. Each algorithm with the different combinations of quantization methods was run 100 times with different sets of randomly generated θ_i and Υ_i, and the results are shown in terms of the median statistic among all experiments. We used such metric to mitigate the effect of outliers. The median is taken considering only the convergent cases for each method across the considered quantization levels. We consider a case to be non-convergent when the ADMM algorithm do not stop before reaching the maximum number of iterations manually set by us. In our experiments, we considered a maximum iteration count of 250 for a network of 10 agents and 300 when considering 30 agents. This set of results considered values of $\eta = 0.2$, $\epsilon = \zeta = 1$, $\rho = 10$, $p = 5$, a tolerance value of $\epsilon_p = 10^{-6}$, $x_i^0 = \bar{z}^0 = \lambda^0 = 0$, and constant $c = 3$ for quantization.

2.5.4.1. Results for 10 agents

Fig. 2.2 (left) shows the results of the median of the 100 experiments for ADMM, STEP-GP and STEP-LGP based methods using the metric presented in Section 2.5.3.3 through the various quantization resolutions tested. The minimum resolution for which any quantization method achieved convergence was 5 bits.

In terms of the LOT metric, STEP-GP presented a better performance in all cases compared to the baseline approaches Sync:UniQuant and Sync:Exact. Also, it can be seen that starting from a resolution of 9 bits the performance of any STEP-LGP based

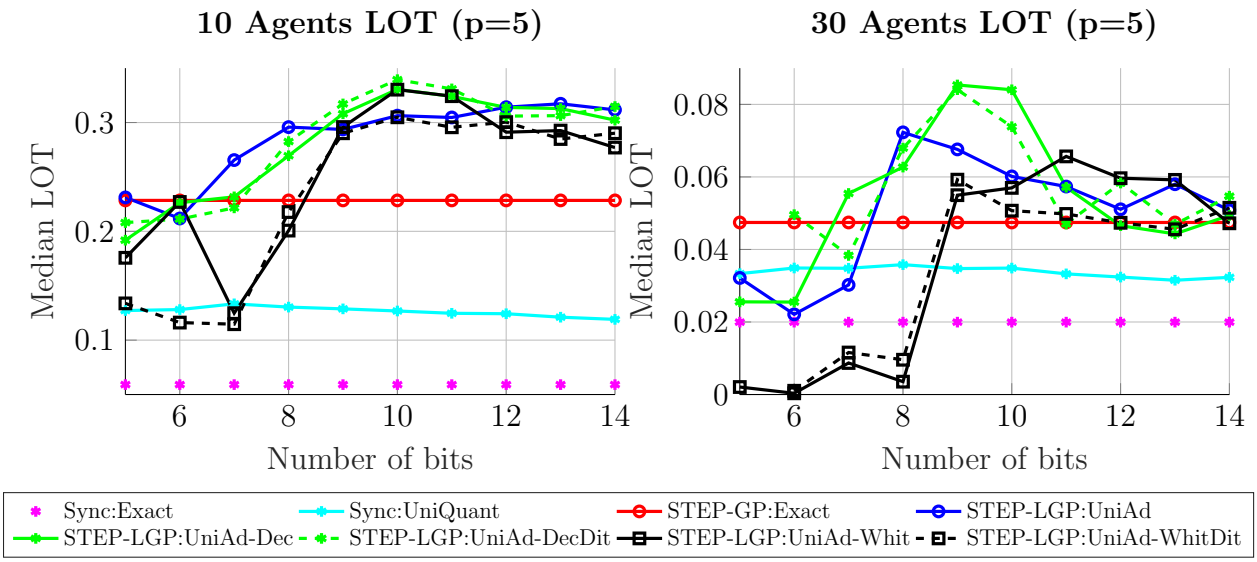

Figure 2.2. Performance in the LOT metric of the adaptive quantization methods at different bit resolutions for 10 agents (left) and 30 agents (right) with $p = 5$. The plots show the median LOT of 100 numerical experiments for different sets of parameters θ_i and Υ_i.

method was better than STEP-GP, Sync:UniQuant, and Sync:Exact, with the peak of performance occurring at 10 bits for STEP-LGP:UniAd-DecDit. For resolutions below 9 bits, STEP-LGP:UniAd outperformed the STEP-GP case starting from 7 bits while STEP-LGP:UniAd-Dec and STEP-LGP:UniAd-DecDit did it starting from 8 bits. For 8 and 7 bits, it is STEP-LGP:UniAd which achieved the best overall performance while STEP-LGP:UniAd-Whit and STEP-LGP:UniAd-WhitDit could not beat the STEP-GP algorithm. Overall, STEP-LGP:UniAd performed consistently good for all the presented resolutions with STEP-LGP:UniAd-Dec and STEP-LGP:UniAd-DecDit presenting the peak of performance starting from a quantization resolution of 9 bits.

2.5.4.2. Results for 30 agents

The performance, in this case, is different than the 10 agents case according to Fig. 2.2 (right) in terms of the LOT metric. It can be seen that STEP-GP presented a better performance in all cases compared to the baseline approaches Sync:UniQuant and

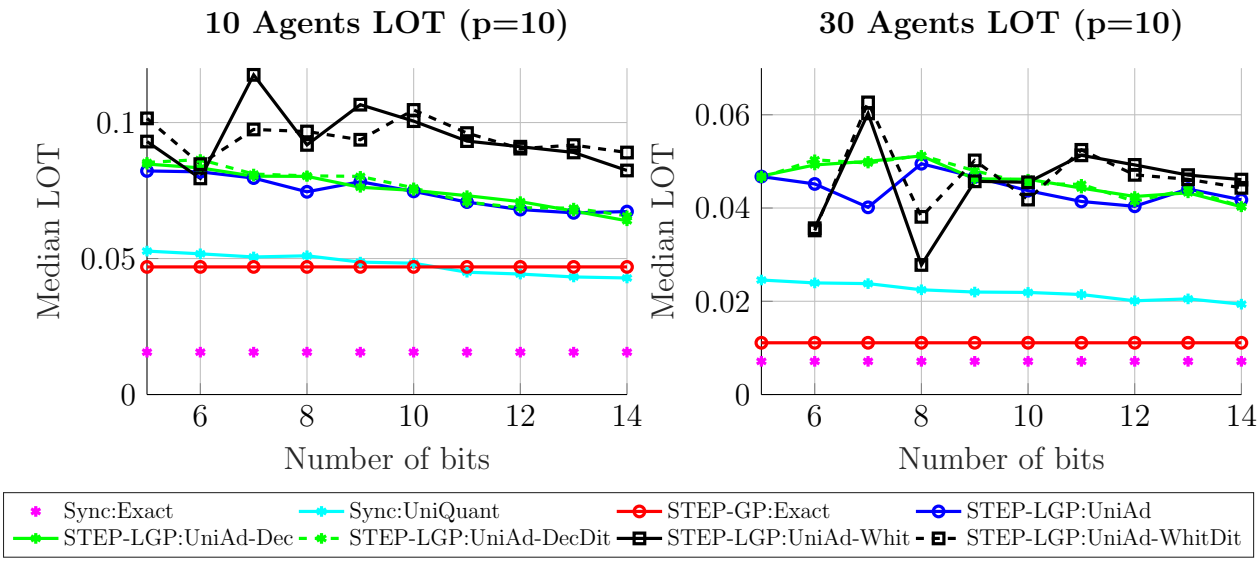

Figure 2.3. Performance in the LOT metric of the adaptive quantization methods at different bit resolutions for 10 agents (left) and 30 agents (right) with $p = 10$. The plots show the median LOT of 100 numerical experiments for different sets of parameters θ_i and Υ_i.

Sync:Exact, however the difference in performance is not as notorious as in the previous case. Similarly to the 10 agents case, STEP-LGP:UniAd-DecDit presented the peak of performance but this time it does for the 9 bits case. Between the 5-8 bits interval, STEP-LGP:UniAd-Whit and STEP-LGP:UniAd-WhitDit could not outperformed STEP-GP, Sync:UniQuant, or Sync:Exact, while the rest of methods using LGP regression always outperformed Sync:Exact and were all able to outperform STEP-GP and Sync:UniQuant starting from the 8 bits case. For 9 and 10 bits, all LGP-based methods presented better performance than STEP-GP with STEP-LGP:UniAd-Dec and STEP-LGP:UniAd-DecDit presenting the better LOT values by a significant margin. Between 11 and 14 bits, the best performance was always attained by a method involving quantization. However, it is noted that the margin between STEP-GP and the methods using LGP regression was significantly reduced compared to the 10 agents case.

2.5.5. Numerical Experiments Results with $p = 10$

In this subsection, we discuss the results for 10 and 30 agents when the dimension of the variables is set to be $p = 10$. The initialization parameters and constant variables considered are the same as in the previous subsection. The corresponding graphs are presented in Figure 2.3.

2.5.5.1. Results for 10 agents

We generated results of the median of 100 numerical experiments for ADMM, STEP-GP and STEP-LGP-based methods using the metric presented in Section 2.5.3.3 through the various quantization resolutions tested. The minimum resolution at which any quantization method achieved convergence was 5 bits.

In terms of the LOT metric, STEP-GP presented a better performance compared to Sync:Exact but it was outperformed by Sync:UniQuant in the cases where such a method had a quantization resolution between 5 and 10 bits. Also, it is observed a stable performance of all the methods using LGP regression through all the quantization resolutions tested as shown in Figure 2.3 (left). In all the cases, those methods consistently beated STEP-GP. The peak of performance was attained by STEP-LGP:UniAd-Whit at 7 bits beating by a small margin its own result for the 9 bits case. Through all the results it is either STEP-LGP:UniAd-Whit or STEP-LGP:UniAd-WhitDit the method that presented the best performance, with the only exception being the 6 bits case. Starting from 10 bits, the methods using whitening presented a significantly better performance compared to all the other methods. Finally, STEP-LGP:UniAd, STEP-LGP:UniAd-Dec, and STEP-LGP:UniAd-DecDit presented a similar behavior through the different quantization

resolutions.

2.5.5.2. Results for 30 agents

Also, we generated the results for 30 agents following the same procedure as in the previous subsection. In Figure 2.3 (right) we can see that the performance, in this case, was similar to the 10 agents case in terms of the LOT metric. The most notorious difference was that STEP-GP was outperformed by Sync:UniQuant for all the tested quantization resolutions. In all the cases, LGP-based methods consistently outperformed STEP-GP. Different from the 10 agents case, the methods STEP-LGP:UniAd-Whit and STEP-LGP:UniAd-WhitDit did not present the same notorious improvement in performance compared to the rest of the methods, however, they still attained the best performance for the 7 bits case.

2.5.6. Overall Remarks

The behavior of methods using whitening transformation reflects that a more complex algorithm can achieve the best results under certain conditions but it lacks the robustness shown (especially at lower quantization bits) by the less complex method STEP-LGP:UniAd. The LGP-based algorithms were able to further reduce the communication expenditure compared to the base STEP-GP algorithm. The best behavior in terms of performance and robustness of any of the proposed quantization-based algorithms is achieved for a resolution greater than 8 bits.

The results showed the potential of our proposed methods to achieve a really good accuracy while significantly reducing the communication cost in comparison to the baseline methods Sync:Exact, Sync:UniQuant, and STEP-GP. Even the less complex proposed

method STEP-LGP:UniAd is good enough for reducing significantly the communication cost while reaching an acceptable accuracy level with consistent performance. The peak of performance in any of the testing scenarios was achieved by a quantization-based method using orthogonal transformation, either Decoupling or whitening.

2.6. Conclusion to Chapter 2

In this chapter, we developed a hybrid approach that combined the Gaussian Process-based learning approach with an adaptive uniform quantization approach to achieve further reduction of the communication cost required in distributed optimization. The resulting quantization error did not follow a Gaussian distribution, so we proposed a new regression algorithm. This algorithm, inspired by GP, resulted in a Linear Minimum Mean Square Estimator named LGP-R, which considered the resulting quantization error statistics. Communication was also reduced by refining the uniform quantizer with an orthogonalization process of the quantizer input to handle the inherent correlation of the quantizer's input components, and with dithering to ensure the uncorrelation between the quantizer's introduced noise and the quantizer's input. Numerical Experiments of a distributed sharing problem showed that our hybrid approaches significantly decreased total communication cost when compared to baseline methods, being able to find the global solution at even low quantization resolutions.

Chapter 3. Convergence Analysis

This chapter presents a convergence analysis of the ADMM algorithm for addressing the sharing problem when applied in conjunction with two algorithms: 1) the stochastic STEP-GP algorithm [23] and 2) its variant named LGP derived in the previous chapter, which includes adaptive uniform quantization. For the case using LGP, the coordinator can assign different quantization resolutions at each iteration, and we assume that the number of bits that can be assigned is unrestricted and can go to infinity. This chapter describes and analyzes the two methods for integrating learning and uniform quantization into the ADMM to reduce its communication overhead and a general formulation of their communication decision method. The problems are formulated for a multi-agent setting.

3.1. Introduction

This chapter serves as a complementary discussion of the derivations presented in Chapter 2. In that chapter, the Alternating Direction Method of Multipliers (ADMM) is used to solve the sharing problem in a multi-agent setting. The main goal is to reduce the ADMM communication overhead. The derived approach named LGP has its foundation in the STEP-GP algorithm presented in [23]. Furthermore, the STEP-GP algorithm extends the work in [22] which proposed an approach called STEP (STructural Estimation of Proximal operator) that relies on the concept of the Moreau Envelope. The STEP approach estimates the unknown gradient of the Moreau Envelope by constructing a set of possible gradients based on past information and then selecting a gradient that is "most likely" the true gradient. The work presented in [23] improved STEP by learning the Moreau envelopes corresponding to the local proximal operators with GP, which are updated online

from previous query data and used to predict the gradient. The resulting algorithm of this work was named STEP-GP.

On the other hand, the work in [23] was extended in the previous chapter (Chapter 2) to consider a uniform quantization on the agent's reply to the coordinator. Following an analysis of the statistical properties of the uniform quantization noise, we were able to derive a mechanism to adapt the uniform quantizer relying on the regression's predicted mean and covariance. This adaptation allows that the quantization error can be approximated to follow a uniform distribution and the correlation of such an error with the quantizer's input can be considered negligible. However, the inclusion of uniform quantization violates the condition for GP where all components are considered Gaussian. For that reason, we derived a new regression scheme constructed upon the concept of a Linear Minimum Mean Square Estimator (LMMSE). To further ensure the conditions for the uncorrelation between the input and error of the quantizer, orthogonal transformation and additive dithering were included. The resulting algorithm was named Linear GP (LGP). These studies present extensive numerical experiments that prove that a significant reduction in communication overhead can be obtained by both algorithms. However, we did not prove the convergence of both algorithms analytically. In this chapter, we complement our previous research by presenting a convergence analysis for STEP-GP and LGP.

Chapter Organization: We initiate this chapter with a summary of key results pertaining to the standard ADMM and the Stochastic Inexact ADMM (SI-ADMM) algorithm that are important for our derivations in Section 3.2. Subsequently, in Section 3.3, we delve into a brief discussion on the learning-integrated ADMM, employing adaptive uniform quantization for the sharing problem. Then, we present the derivation of a conver-

gence proof for the STEP-GP algorithm in Section 3.4, where we prove that the expected value of the ADMM residual goes to zero as the algorithmic iterations go to infinity and do so at a geometric rate. A similar conclusion is reached in Section 3.5, where we present a convergence analysis for the LGP algorithm assuming that the quantization resolution can be varied and its variation is unbounded. The convergence analysis for the LGP algorithm when the quantization resolution is bounded is presented in Section 3.6, where it is shown that the expectation of the ADMM residual is bounded. We present in Section 3.7 the connection between the derived convergence analysis and the LGP algorithm as defined in Chapter 2, since the results in that chapter consider fixed quantization. Finally, the conclusions for this chapter are presented in Section 3.8.

3.2. Preliminary Convergence Results

3.2.1. Generalized ADMM Convergence Analysis

This subsection summarizes useful results from [44]; however, because the notation used in [44] is different from that used in [23] and Chapter 2, it will be adjusted to match our notations. The results from [44] are for the generalized ADMM algorithm solving the general problem:

$$\begin{aligned} \text{minimize}_{x,y} \quad & f(x) + h(y) \\ \text{subject to} \quad & Ax + By = c \end{aligned}$$

The algorithm is different from the standard ADMM by introducing smoothing terms based on the norms. Let the augmented Lagrangian be

$$\mathcal{L}(x, y, m) = f(x) + h(y) - m^\top (Ax + By - c) + \frac{1}{2\rho} \|Ax + By - c\|_2^2$$

Choose $Q \succeq 0$ and a symmetric matrix P (possibly indefinite). Then each algorithm's iteration consists of:

$$y^{k+1} = \text{argmin}_y \mathcal{L}(x^k, y, m^k) + \frac{1}{2}\|y - y^k\|_Q^2$$
$$x^{k+1} = \text{argmin}_x \mathcal{L}(x, y^{k+1}, m^k) + \frac{1}{2}\|x - x^k\|_P^2$$
$$m^{k+1} = m^k - \frac{\zeta}{\rho}(Ax^{k+1} + By^{k+1} - c).$$

Note that the standard ADMM is a special case of the generalized ADMM where $P = Q = 0$ and $\zeta = 1$. Let $s = [x, y, m]$ with corresponding versions s^* for the optimal solutions and s^k for the algorithm iterations. In addition, define the following matrices:

$$\hat{P} = P + (1/\rho)A^\top A, \qquad G = \begin{bmatrix} \hat{P} & & \\ & Q & \\ & & \frac{\rho}{\zeta} I_p \end{bmatrix}$$

where p is the dimension of m. For standard ADMM, with $P = Q = 0$ and $\zeta = 1$, we have

$$\hat{P} = \beta A^\top A, \qquad G = \begin{bmatrix} \beta A^\top A & & \\ & 0 & \\ & & \rho I_p \end{bmatrix} = (1/\rho) G_0^\top G_0, \qquad G_0 = \begin{bmatrix} A & & \\ & 0 & \\ & & \rho I_p \end{bmatrix}$$

Also define the norm $\|s\|_G = \sqrt{s^\top G s}$. Assumptions 1 and 2 in [44] are standard for ADMM convergence.

- **Assumption 1**: There exists a saddle point $s^* = (x^*, y^*, m^*)$ to the problem, namely, x^*, y^*, and m^* satisfying the KKT conditions:

$$A^\top m^* \in \partial f(x^*)$$
$$B^\top m^* \in \partial h(y^*)$$
$$Ax^* + By^* - c = 0.$$

- **Assumption 2**: Functions f and h are convex. One of them is also strongly convex.

Table 1 Four scenarios leading to linear convergence

Scenario	Strongly convex	Lipschitz continuous	Full row rank	Additional assumptions
1	f	∇f	A	*If* $Q \succ 0$, B has full column rank
2	f, g	∇f	A	
3	f	$\nabla f, \nabla g$	–	B has full column rank
4	f, g	$\nabla f, \nabla g$	–	

Table 2 Summary of linear convergence results

Case	P, $\hat{P}$	Q	Any scenario 1–4	
			Q-linear convergence	R-linear convergence
1	$P = 0$	$= 0$	(Ax^k, λ^k)	x^k, $(y^k$ or $By^k)^*$, λ^k
2	$\hat{P} \succ 0$	$= 0$	(x^k, λ^k)	
3	$P = 0$	$\succ 0$	(Ax^k, y^k, λ^k)	
4	$\hat{P} \succ 0$	$\succ 0$	(x^k, y^k, λ^k)	

Column rank of B; otherwise, only By^k has R-linear convergence
* In cases 1 and 2, scenario 1, R-linear convergence of y^k requires full

Under these assumptions and another technical assumption, Theorem 3.4 in [44] provides a bound on the convergence rate of generalized ADMM. The theorem is reproduced below.

Theorem 3 (Theorem 3.4 in [44]) *Assume Assumptions 1 and 2,* $\zeta = 1$*, and that* s^k *of the Generalized ADMM is bounded (see remark below). For all scenarios in Table 1, there exists* $\delta > 0$ *such that*

$$\|s^k - s^*\|_G^2 \geqslant (1+\delta)\|s^{k+1} - s^*\|_G^2$$

Remark 1 *In terms of the boundedness of* $\{s^k\}$*, Remark 2.2 in [44] provides several conditions. For example, if the objective functions are coercive then the boundedness is guaranteed. Also, for the standard ADMM, the boundedness is guaranteed if* A *and* B *have full column rank.*

We now apply the above results to standard ADMM and more specifically the sharing problem, as defined in [8, 10] having the form

$$\text{minimize} \quad \sum_{i=1}^{n} f_i(x_i) + h\left(\sum_{i=1}^{n} x_i\right). \tag{3.1}$$

Theorem 2.2 in [44] states the convergence of $\{s^k\}$ to the KKT point. In particular, for the standard ADMM as a special case of the generalized ADMM, under the same assumptions as above, we have $m^k \to m^*$, $Ax^k \to Ax^*$, and $By^k \to By^*$. The sharing problem is a special case of the standard ADMM problem with $A = I$ and $B = -I$; therefore, we have $u^k \to u^*$, $x_i^k \to x_i^*$, and $\bar{y}^k \to \bar{y}^*$ (or $y_i^k \to y_i^*$). Note that here u is just the scaled version of the dual variables m.

Applying Theorem 3 to the standard ADMM and the sharing problem, we have:

- For standard ADMM: there exists $\delta > 0$ such that
$$\left\| \begin{bmatrix} A(x^k - x^*) \\ u^k - u^* \end{bmatrix} \right\|_2^2 \geqslant (1+\delta) \left\| \begin{bmatrix} A(x^{k+1} - x^*) \\ u^{k+1} - u^* \end{bmatrix} \right\|_2^2$$

- For the sharing problem with standard ADMM: there exists $\delta > 0$ such that
$$\left\| \begin{bmatrix} x_i^k - x_i^* \\ u^k - u^* \end{bmatrix} \right\|_2^2 \geqslant (1+\delta) \left\| \begin{bmatrix} x_i^{k+1} - x_i^* \\ u^{k+1} - u^* \end{bmatrix} \right\|_2^2,$$
for all i stacked vertically.

3.2.2. Stochastic inexact ADMM (SI-ADMM) Convergence Analysis

This subsection summarizes the stochastic inexact ADMM for the general ADMM problem and its convergence result in [45]. The paper considers the general stochastic ADMM problem (of which the sharing problem is a special case):

$$\begin{aligned} \text{minimize}_{x,y} \quad & \mathbb{E}[\tilde{f}(x,\xi)] + \mathbb{E}[\tilde{h}(y,\xi)] \\ \text{subject to} \quad & Ax + By = c \end{aligned}$$

for some random variable ξ with known distribution. This problem can be solved by the standard ADMM if $f(x) = \mathbb{E}[\tilde{f}(x,\xi)]$ and $h(y) = \mathbb{E}[\tilde{h}(y,\xi)]$ can be calculated analytically and easily. However, this is not true in many cases and $f(x)$ and $g(y)$ can only be approximated. This means that the ADMM steps where the proximal operators of f

and g are evaluated cannot be done exactly. For example, $\text{argmin}_x f(x) + \frac{1}{2\rho}\|x - z_k\|_2^2$ cannot be solved exactly easily. To overcome this issue, the paper proposes applying a sampled gradient descent approach to solve the proximal minimization problems. For instance, given a sequence of N_k^x samples $\{\xi_{k,1}^x, \ldots, \xi_{k,N_k^x}^x\}$ of ξ, then the above stochastic proximal minimization can be solved approximately by iterative gradient descent steps: $x_{k+1}^{i+1} = x_{k+1}^i - \gamma \nabla_x \left(\mathbb{E}[\tilde{f}(x, \xi_{k,i}^x)] + \frac{1}{2\rho}\|x - v_k\|_2^2 \right)$. It can be shown that as N_k^x increases, the error between $x_{k+1}^{N_k^x}$ and the true solution $x_{k+1}^\star$ decreases and can be bounded.

Given the above approach, the stochastic ADMM algorithm is modified as follows (simplified version of Algorithm 2 SI-ADMM in [45]):

$$
\begin{aligned}
y_{k+1} \quad & \text{is such that } \mathbb{E}[\|y_{k+1} - y_{k+1}^*\|^2] \leqslant \eta_{k+1} \\
x_{k+1} \quad & \text{is such that } \mathbb{E}[\|x_{k+1} - \tilde{x}_{k+1}^*\|^2] \leqslant \eta_{k+1} \\
m_{k+1} \quad & = m_k - \gamma\rho(Ax_{k+1} + By_{k+1} - c)
\end{aligned}
$$

where y_{k+1}^* is the unknown true solution of the generalized proximal minimization (with an extra smoothing term), $\tilde{x}_{k+1}^*$ is the unknown true solution of the generalized proximal minimization that uses y_{k+1} instead of y_{k+1}^* (hence the tilde).

The key theorem of the papers can be stated below (Theorem 2 in [45]).

Theorem 4 *Consider the above SI-ADMM algorithm. Under certain technical assumptions (see [45]) and* $\sum_{k=1}^\infty \sqrt{\eta_k} < \infty$*, we have that* $\|s_k - s^*\|_G \to 0$ *almost surely as* $k \to \infty$*.*

Here, $s = [x, y, u]$ and G is a matrix derived in the work on generalized ADMM [44].

Lemma 3 *Let* $\{v_k\}$ *be a sequence of nonnegative random variables, where* $\mathbb{E}[v_0] < \infty$*, and*

let $\{u_k\}$ *and* $\{\mu_k\}$ *be deterministic scalar sequences such that:*

$$\mathbb{E}[v_{k+1}|v_{0,\ldots,v_k}] \leqslant (1-u_k)v_k + \mu_k \ \ a.s. \ \forall k \geqslant 0,$$

$$0 \leqslant u_k \leqslant 1, \quad \mu_k \geqslant 0, \quad \forall k \geqslant 0, \quad \textstyle\sum_{k=0}^{\infty} u_k = \infty, \quad \sum_{k=0}^{\infty} \mu_k < \infty, \quad \lim_{k\to\infty} \frac{\mu_k}{u_k} = 0.$$

Then $v_k \to 0$ *almost surely as* $k \to \infty$.

3.3. Problem Formulation

In the setting of a multi-agent optimization problem where the structure resembles the sharing problem as defined in (3.1), each of the n agents has local decision variables $x_i \in \mathbb{R}^p$ and a strongly convex local cost function f_i: $\mathbb{R}^p \mapsto \mathbb{R}$. Their objective is to minimize the overall system cost, which comprises their local costs and a convex shared global cost function h: $\mathbb{R}^p \mapsto \mathbb{R}$.

The problem presented in (3.1) can be equivalently reformulated by introducing auxiliary variables y_i for each x_i as

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n} f_i(x_i) + h\left(\sum_{i=1}^{n} y_i\right) \\ \text{subject to} \quad & x_i - y_i = 0, \quad \forall i = 1, \ldots, N. \end{aligned} \tag{3.2}$$

Because the agents must keep their local cost function f_i private, each agent i only provides the solution to the following local *proximal minimization problem* to the coordinator

$$\mathbf{prox}_{\frac{1}{\rho}f_i}(z_i^k) = \underset{x_i\in\mathbb{R}^p}{\arg\min} \left\{ f_i(x_i) + \frac{\rho}{2}\|x_i - z_i^k\|^2 \right\}, \tag{3.3}$$

in response to a value (a query) z_i^k sent to it by the coordinator at iteration k, where $\rho > 0$ is a penalty parameter. The problem posed in (3.2) can be solved using the ADMM algorithm following the query response mechanism and ADMM updates explained in Section 2.1.

Our approach named LGP considers adding adaptive uniform quantization as explained in Chapter 2. We assume in the following results that the adaptation of the quantizer is done not only over the middle point and the window length but also over the quantization resolution. This means that the coordinator can adjust the bits used for quantization as needed. The LGP approach considers first defining the communication decision variable for agent i in iteration k as

$$\gamma_i^k = \begin{cases} 1, & \text{if agent } i \text{ is queried} \\ 0, & \text{otherwise.} \end{cases}$$

When $\gamma_i^k = 1$, the query z_i^k is sent to agent i to obtain the quantized value of $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$. In contrast, when $\gamma_i^k = 0$, we use the predicted value $\mu_i^k(z_i^k)$ given by the LGP regression. We then define an expression β_i^k as

$$\beta_i^k = \gamma_i^k \mathbb{Q}_{ua}(\nabla f_i^{\frac{1}{\rho}}(z_i^k)) + (1 - \gamma_i^k)\mu_i^k(z_i^k).$$

Contrary to the regular STEP-GP algorithm, we always have a source of inexactness either from the LGP prediction when there is no query or from the adaptive uniform quantization when a query is made. This is due to the decision mechanism used in Chapter 2, where we aim to limit the general inexactness by ψ^k. The value of the threshold ψ^k determines which agents to be queried. This decision mechanism is expressed in the following

optimization problem:

$$\begin{aligned} \underset{\gamma^k, b^k}{\text{minimize}} \quad & \sum_{i=1}^{n} \left[(\gamma_i^k) b_i^k \right] \\ \text{subject to} \quad & b_i^k \in \mathcal{N}, \\ & \gamma_i^k \in \{0, 1\}, \\ & \sum_{i=1}^{n} \left[\gamma_i^k \frac{\theta}{2^{2b_i^k}} \text{trace}(\Sigma_i^k(z_i^k)) + (1 - \gamma_i^k) \text{trace}(\Sigma_i^k(z_i^k)) \right] < \psi^k. \end{aligned} \tag{3.4}$$

The threshold ψ^k decreases at each iteration to keep up with the decrease of $\sum_{i=1}^{n} \text{trace}(\Sigma_i^k(z_i^k))$. In addition, it is important to note that $\gamma_i^k \frac{\theta}{2^{2b_i^k}} < (1 - \gamma_i^k)$ and, depending on the value of b_i^k, the uncertainty resulting from the prediction could be much greater than the uncertainty resulting from the quantization of a particular agent. Furthermore, b_i^k is assumed to be unbounded, so its value can be as large as necessary to satisfy the constraint. Thus, the value of this variable can go to infinity, making the quantization uncertainty vanish if necessary. Finally, the sharing ADMM expression considering the communication reduction can be expressed as:

$$\begin{aligned} \bar{y}^{k+1} &= \underset{\bar{y} \in \mathbb{R}^p}{\arg\min} \left\{ h(n\bar{y}) + (n\rho/2) \|\bar{y} - \bar{x}^k - u^k\|^2 \right\} \\ x_i^{k+1} &= z_i^k - (1/\rho)\beta_i^k \\ u^{k+1} &= u^k + \bar{x}^{k+1} - \bar{y}^{k+1}. \end{aligned} \tag{3.5}$$

3.4. Convergence Proof of the STEP-GP algorithm

In this section, we present a convergence analysis for the STEP-GP algorithm presented in [23]. This algorithm has ADMM updates similar to (3.5) when solving the shar-

ing problem. However, it considers the following problem for querying decision-making

$$\begin{aligned} \underset{\gamma^k}{\text{minimize}} \quad & \|\gamma^k\|_1 \\ \text{subject to} \quad & \gamma_i^k \in \{0,1\}. \\ & \sum_{i=1}^{n} \left[(1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k))\right] < \psi^k. \end{aligned} \tag{3.6}$$

Here, we want to minimize the number of communicating agents while keeping the global prediction uncertainty bounded. The threshold ψ^k decreases at each iteration, ensuring that the uncertainty given to the system also reduces over time until it eventually vanishes. However, at the moment when the threshold ψ^k decreases too much, we query all agents at each iteration impacting the communication reduction in the last rounds before reaching convergence.

3.4.1. Preliminaries

Define $s^k = [\bar{x}^k; \bar{y}^k; u^k]$ and that $\mathcal{I}_i^k$ collects the query information from each agent i up to iteration k. The STEP-GP algorithm defines the mapping $\Gamma^{k+1} : s^k \to s^{k+1}$ which gives a mixture of inexact and exact values depending on the value of the decision variable γ_i^k. On the other hand, the exact ADMM algorithm defines the exact mapping $\Gamma_*^{k+1} : s^k \to s_*^{k+1}$ where $s_*^{k+1} = [\bar{x}_*^k; \bar{y}_*^k; u_*^k]$ are the exact values. Note that $\bar{y_*}^k = \bar{y}^k$, therefore it is always known exactly.

The following convergence proof is constructed upon the querying policy in (3.6) and the mean square error between the inexact and exact values of x_i^{k+1} and u^{k+1}. Those expressions are given by:

- We know that $(x_i^{k+1} - x_{*,i}^{k+1}) = (1/\rho)(\beta_i^{k+1} - \beta_{*,i}^{k+1})$, and it can be shown that $\mathbb{E}[||x_i^{k+1} - x_{*,i}^{k+1}||^2|\gamma_i^k] = \text{trace}(\text{Cov}(x_i^{k+1} - x_{*,i}^{k+1}|\gamma_i^k)) = (1/\rho)^2((1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k)))$.

Thus,

$$\mathbb{E}[||x^{k+1} - x_*^{k+1}||^2|\gamma^k] = (1/\rho)^2 \sum_{i=1}^{n} \left((1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k))\right)$$

- We can express $u^{k+1} = -(1/\rho)\bar{\beta}^k$, so

$$\mathbb{E}[||u^{k+1} - u_*^{k+1}||^2|\gamma^k] = \left(\frac{(1/\rho)}{n}\right)^2 \sum_{i=1}^{n} \left((1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k))\right) \tag{3.7}$$

- Due to the querying policy defined in problem (3.6), we have that

$$\mathbb{E}[||x^{k+1} - x_*^{k+1}||^2|\gamma^k] \leq (1/\rho)^2\psi^k$$

3.4.2. Upper-bound on the expected value of the ADMM residual for STEP-GP

In each iteration, we consider the following variables: s^k is the state of the algorithm at the beginning; s^{k+1} is the output of the STEP-GP algorithm, which is a random variable; s_*^{k+1} is implicitly produced by the exact ADMM algorithm. Therefore, the STEP-GP algorithm produces a sequence of random variable samples $\{s^k\}$.

Let s^* be the KKT solution (to which the exact ADMM converges). Note that s^* is a fixed point of the mapping Γ^*, that is, $s^* = \Gamma^*(s^*)$. Define $\hat{s}^k = [x^k; u^k]$, which is part of s^k (excluding $\bar{y}^k$). We consider the residual $\epsilon^k = ||\hat{s}^k - \hat{s}^*||_2 = ||s^k - s^*||_G$. Henceforth, we will omit the conditioning on γ^k for brevity. Let $\mathcal{I}^k$ denote the total information collected, i.e., the total history of the queries, of all agents i up to iteration k; in other words, $\mathcal{I}^k = \bigcup_i \mathcal{I}_i^k$.

Theorem 5 *Consider the STEP-GP algorithm for the sharing problem. Suppose that the 3 assumptions in [45] hold and* $\sum_{i=1}^{\infty} \sqrt{\psi^k} < \infty$. *Then* $\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k]$ *is bounded by*

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] \leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\epsilon^k|\mathcal{I}^{k-1}],$$

where $g = (1/\rho)\sqrt{(1 + \left(\frac{1}{n^2}\right))}$.

Proof: This proof follows the proof of Theorem 2 in [45]. First, we develop a bound on $\mathbb{E}[||s^{k+1} - s^*||_G]$.

$$\mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] = \mathbb{E}\left[\sqrt{\sum_{i=1}^{n} ||x_i^{k+1} - x_{*,i}^{k+1}||_2^2 + ||u^{k+1} - u_*^{k+1}||_2^2}\middle|\mathcal{I}^k\right]$$

$$\leq \sqrt{\sum_{i=1}^{n} \mathbb{E}[||x_i^{k+1} - x_{*,i}^{k+1}||_2^2|\mathcal{I}^k] + \mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k]}$$

$$= \sqrt{(1/\rho)^2\psi^k + \mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k]}$$

where the inequality comes from applying Jensen's inequality, the concavity of the square root, and the querying policy condition. For the second term we apply (3.7) and the constraint in (3.4), giving that

$$\mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k] \leq \left(\frac{(1/\rho)}{n}\right)^2 \psi^k.$$

Therefore,

$$\mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] \leq \sqrt{(1/\rho)^2\psi^k + \left(\frac{(1/\rho)}{n}\right)^2 \psi^k} = g\sqrt{\psi^k},$$

where $g = (1/\rho)\sqrt{(1 + \left(\frac{1}{n^2}\right))}$.

Now, consider the residual ϵ^k. We have that

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] = \mathbb{E}[||s^{k+1} - s^*||_G|\mathcal{I}^k] = \mathbb{E}[||s^{k+1} - s_*^{k+1} + s_*^{k+1} - s^*||_G|\mathcal{I}^k]$$

$$\leq \mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] + \mathbb{E}[||\Gamma^*(s^k) - \Gamma^*(s^*)||_G|\mathcal{I}^k]$$

$$\leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[||s^k - s^*||_G|\mathcal{I}^k]$$

for some $\delta > 0$, where the last inequality comes from Theorem 3 in [44]. It follows that

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] \leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\epsilon^k|\mathcal{I}^{k-1}].$$

□

3.4.3. Rate of Convergence and Convergence of the Expected Value of the Residual of STEP-GP

In the following lines, we prove that if ψ^k decreases geometrically, then the expected value of the mean square error of the ADMM residual when using the STEP-GP algorithm converges to zero as $k \to \infty$ and does so at a geometric rate. First, we restate Lemma 4 proven in [45] as:

Lemma 4 *Given a function $f(z) = zw^z$ where $w < 1$. Then, for all $z \geq 0$, we have that*

$$zw^z < Dl^z,$$

where $w < l < 1$ and $D > \frac{1}{\ln(l/w)^e}$.

This lemma makes the following Theorem to hold:

Theorem 6 *Consider the STEP-GP algorithm. Suppose that Theorem 5 holds and $\sqrt{\psi^k} = (\alpha)^k$ for some $0 < \alpha < 1$ (note that $(\alpha)^k$ refers to a constant raised to the power k). Then for every $k > 0$, we have that*

$$\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \leq (gD + \varepsilon^0)(l)^k,$$

where $l > r \triangleq \max(\frac{1}{\sqrt{1+\delta}}, \alpha)$ and D is chosen such that $D > \frac{1}{e\ln(l/r)}$. Furthermore, $\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \to 0$ as $k \to \infty$.

Proof: Let $a = \frac{1}{\sqrt{1+\delta}}$. Since $\sqrt{\psi^k} = (\alpha)^k$ where $\alpha < 1$, we have the following

sequence of inequalities based on the bound $\mathbb{E}[\varepsilon^{k+1}|\mathcal{I}^k] \leqslant g(\alpha)^k + a\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}]$.

$$\begin{aligned}
\mathbb{E}[\varepsilon^{k+1}|\mathcal{I}^k] &\leqslant a\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] + g(\alpha)^k \leqslant (a)^2\mathbb{E}[\varepsilon^{k-1}|\mathcal{I}^{k-2}] + ag(\alpha)^{k-1} + g(\alpha)^k \\
&\leqslant (a)^3\mathbb{E}[\varepsilon^{k-2}|\mathcal{I}^{k-3}] + (a)^2 g(\alpha)^{k-2} + ag(\alpha)^{k-1} + g(\alpha)^k \\
&\vdots \\
&\leqslant (a)^{k+1}\varepsilon^0 + g\sum_{j=0}^{k}(a)^{k-j}(\alpha)^j \leqslant (r)^k\varepsilon^0 + c\sum_{j=0}^{k}(r)^k \\
&= (\varepsilon^0 + g(k+1))(r)^k \leqslant (\varepsilon^0 + ck)(r)^k \\
\Rightarrow \mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] &\leqslant (\varepsilon^0 + g(k-1))(r)^{k-1}.
\end{aligned}$$

From Lemma 4, it can be shown that there exist scalars l and D satisfying $l \in (r, 1)$ and $D > 1/\ln((l/r)^e)$ such that

$$\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \leqslant \varepsilon^0(r)^{k-1} + g(k-1)(r)^{k-1} < \varepsilon^0(r)^{k-1} + gD(r)^{k-1} < (\varepsilon^0 + gD)(l)^{k-1}.$$

Finally, since $l < 1$, it follows that as $k \to \infty$ then $\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \to 0$. □

3.5. Convergence Proof of the LGP algorithm with Unbounded Quantization Resolution

In this section, we present a convergence analysis for the LGP algorithm presented in Chapter 2. However, we consider the case where the coordinator can vary the quantization resolution at each iteration and it is not fixed as in our previous study. Moreover, we assume that we can assign an infinitely large quantization resolution if needed.

3.5.1. Preliminaries

The mapping and definitions of s^*, s_*^k, and s^k are the same as in Section 3.4.1.

The convergence proof is constructed upon the querying policy in (3.4) and the mean square error between the exact and inexact values of x_i^{k+1} and u^{k+1}. These expressions are given by:

- We know that $(x_i^{k+1} - x_{*,i}^{k+1}) = (1/\rho)(\beta_i^{k+1} - \beta_{*,i}^{k+1})$, and it can be shown that $\mathbb{E}[||x_i^{k+1} - x_{*,i}^{k+1}||^2|\gamma_i^k] = \text{trace}(\text{Cov}(x_i^{k+1} - x_{*,i}^{k+1}|\gamma_i^k)) = (1/\rho)^2(\gamma_i^k \frac{\theta}{2^{2b_i^k}}\text{trace}(\Sigma_i^k(z_i^k)) + (1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k)))$. Thus,

$$\mathbb{E}[||x^{k+1} - x_*^{k+1}||^2|\gamma^k] = (1/\rho)^2 \sum_{i=1}^{n} \left(\gamma_i^k \frac{\theta}{2^{2b_i^k}}\text{trace}(\Sigma_i^k(z_i^k)) + (1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k)) \right)$$

- We can express $u^{k+1} = -(1/\rho)\bar{\beta}^k$, so

$$\mathbb{E}[||u^{k+1} - u_*^{k+1}||^2|\gamma^k] = \left(\frac{(1/\rho)}{n}\right)^2 \sum_{i=1}^{n} \left(\gamma_i^k \frac{\theta}{2^{2b_i^k}}\text{trace}(\Sigma_i^k(z_i^k)) + (1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k)) \right) \tag{3.8}$$

- Due to the querying policy defined in the problem (3.4), we have that

$$\mathbb{E}[||x^{k+1} - x_*^{k+1}||^2|\gamma^k] \leq (1/\rho)^2 \psi^k$$

3.5.2. Upper-bound on the expected value of the ADMM residual for LGP

In each iteration, we consider the following variables: s^k is the state of the algorithm at the beginning; s^{k+1} is the output of the LGP algorithm, which is a random variable; s_*^{k+1} is implicitly produced by the exact ADMM algorithm. Therefore, the LGP algorithm produces a sequence of random variable samples $\{s^k\}$.

Let s^* be the KKT solution (to which the exact ADMM converges). Note that s^* is a fixed point of the mapping Γ^*, that is, $s^* = \Gamma^*(s^*)$. Define $\hat{s}^k = [x^k; u^k]$, which is part of s^k (excluding $\bar{y}^k$). We will consider the residual $\epsilon^k = ||\hat{s}^k - \hat{s}^*||_2 = ||s^k - s^*||_G$. Henceforth, we will omit the conditioning on γ^k for the sake of brevity. Let $\mathcal{I}^k$ denote the total information collected, i.e., the total history of the queries, of all agents i up to iteration k; in other words, $\mathcal{I}^k = \bigcup_i \mathcal{I}_i^k$.

Theorem 7 *Consider the LGP algorithm for the sharing problem. Suppose that the 3*

assumptions in [45] hold and $\sum_{k=1}^{\infty}\sqrt{\psi^k} < \infty$. *Then* $\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k]$ *is bounded by*

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] \leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\epsilon^k|\mathcal{I}^{k-1}],$$

where $g = (1/\rho)\sqrt{(1+\left(\frac{1}{n^2}\right))}$.

Proof: This proof follows the proof of Theorem 2 in [45]. We first develop a bound on $\mathbb{E}[||s^{k+1} - s^*||_G]$.

$$\mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] = \mathbb{E}\left[\sqrt{\sum_{i=1}^{n}||x_i^{k+1} - x_{*,i}^{k+1}||_2^2 + ||u^{k+1} - u_*^{k+1}||_2^2}\middle|\mathcal{I}^k\right]$$

$$\leq \sqrt{\sum_{i=1}^{n}\mathbb{E}[||x_i^{k+1} - x_{*,i}^{k+1}||_2^2|\mathcal{I}^k] + \mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k]}$$

$$= \sqrt{(1/\rho)^2\psi^k + \mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k]}$$

where the inequality comes from applying Jensen's inequality, the concavity of the square root, and the querying policy condition. For the second term we apply (3.8) and the constraint in (3.4), giving that

$$\mathbb{E}[||u^{k+1} - u_*^{k+1}||_2^2|\mathcal{I}^k] \leq \left(\frac{(1/\rho)}{n}\right)^2\psi^k$$

. Therefore,

$$\mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] \leq \sqrt{(1/\rho)^2\psi^k + \left(\frac{(1/\rho)}{n}\right)^2\psi^k} = g\sqrt{\psi^k},$$

where $g = (1/\rho)\sqrt{(1+\left(\frac{1}{n^2}\right))}$.

Now, consider the residual ϵ^k. We have that

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] = \mathbb{E}[||s^{k+1} - s^*||_G|\mathcal{I}^k] = \mathbb{E}[||s^{k+1} - s_*^{k+1} + s_*^{k+1} - s^*||_G|\mathcal{I}^k]$$

$$\leq \mathbb{E}[||s^{k+1} - s_*^{k+1}||_G|\mathcal{I}^k] + \mathbb{E}[||\Gamma^*(s^k) - \Gamma^*(s^*)||_G|\mathcal{I}^k]$$

$$\leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[||s^k - s^*||_G|\mathcal{I}^k]$$

for some $\delta > 0$, where the last inequality comes from Theorem 3 in [44]. It follows that

$$\mathbb{E}[\epsilon^{k+1}|\mathcal{I}^k] \leq g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\epsilon^k|\mathcal{I}^{k-1}].$$

□

3.5.3. Rate of Convergence and Convergence of the Expected Value of the Residual for LGP

In the following lines, we prove that if ψ^k decreases geometrically, then the expected value of the mean square error of the ADMM residual when using the unbounded LGP algorithm converges to zero as $k \to \infty$ and does so at a geometric rate. Taking into account Lemma 4, we formulate the following theorem:

Theorem 8 *Consider the LGP algorithm. Suppose that* $\mathbb{E}[\varepsilon^{k+1}|\mathcal{I}^k] \leqslant g\sqrt{\psi^k} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}]$ *holds and* $\sqrt{\psi^k} = (\alpha)^k$ *for some* $0 < \alpha < 1$ *(note that* $(\alpha)^k$ *refers to a constant raised to the power* k*). Then for every* $k > 0$*, we have that*

$$\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \leq (gD + \varepsilon^0)(l)^k,$$

where $l > r \triangleq \max(\frac{1}{\sqrt{1+\delta}}, \alpha)$ *and* D *is chosen such that* $D > \frac{1}{e\ln(l/r)}$*. Furthermore,* $\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \to 0$ *as* $k \to \infty$*.*

Proof: Let $a = \frac{1}{\sqrt{1+\delta}}$. Since $\sqrt{\psi^k} = (\alpha)^k$ where $\alpha < 1$, we have the following

sequence of inequalities based on the bound $\mathbb{E}[\varepsilon^{k+1}|\mathcal{I}^k] \leqslant g(\alpha)^k + a\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}]$.

$$\begin{aligned}
\mathbb{E}[\varepsilon^{k+1}|\mathcal{I}^k] &\leqslant g\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] + g(\alpha)^k \leqslant (a)^2\mathbb{E}[\varepsilon^{k-1}|\mathcal{I}^{k-2}] + ag(\alpha)^{k-1} + c(\alpha)^k \\
&\leqslant (a)^3\mathbb{E}[\varepsilon^{k-2}|\mathcal{I}^{k-3}] + (a)^2 g(\alpha)^{k-2} + ag(\alpha)^{k-1} + g(\alpha)^k \\
&\vdots \\
&\leqslant (a)^{k+1}\varepsilon^0 + g\sum_{j=0}^{k}(a)^{k-j}(\alpha)^j \leqslant (r)^k\varepsilon^0 + g\sum_{j=0}^{k}(r)^k \\
&= (\varepsilon^0 + g(k+1))(r)^k \leqslant (\varepsilon^0 + gk)(r)^k \\
\Rightarrow \mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] &\leqslant (\varepsilon^0 + g(k-1))(r)^{k-1}.
\end{aligned}$$

From Lemma 4, it can be shown that there exist scalars l and D satisfying $l \in (r, 1)$ and $D > 1/\ln((l/r)^e)$ such that

$$\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \leqslant \varepsilon^0(r)^{k-1} + g(k-1)(r)^{k-1} < \varepsilon^0(r)^{k-1} + gD(l)^{k-1} < (\varepsilon^0 + gD)(l)^{k-1}.$$

Finally, since $l < 1$, it follows that as $k \to \infty$ then $\mathbb{E}[\varepsilon^k|\mathcal{I}^{k-1}] \to 0$. □

3.6. Convergence Analysis of the LGP algorithm with Bounded Quantization Resolution

The convergence analysis in the previous subsection assumes that the quantization resolution can be infinitely large, eventually making the uncertainty zero if all agents communicate. In real life, having an infinite quantization resolution defies the purpose of using quantization. However, not allowing the quantization resolution to be infinitely large goes against the condition to conclude the convergence that the uncertainty eventually reaches zero as k goes to infinity.

Let us define the maximum possible value of b_i^k as $b_{\max}$. In the limit case, let us say

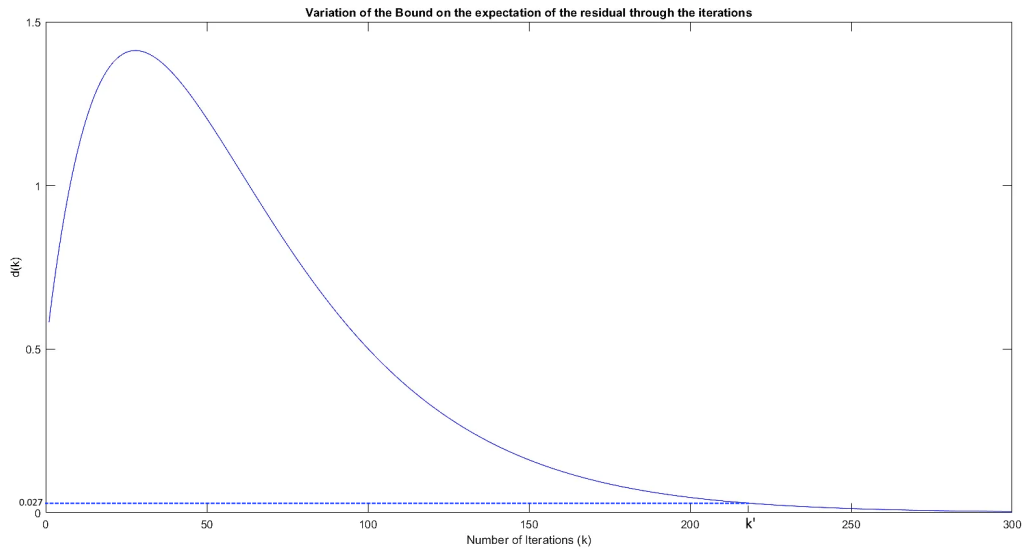

Figure 3.1. Upper bound of the expected value of the residual through the iterations for values of $\alpha = 0.97$, $n = 10$, $\rho = 10$.

that iteration k' is the last iteration where the constraint is met having the form:

$$\sum_{i=1}^{n} \left(\frac{\theta}{2^{2b_{\max}}} \text{trace}(\Sigma_i^{k'}(z_i^{k'})) \right) < \psi_i^{k'},$$

where we assume the lowest possible uncertainty that is attained when all agents are queried. This is because for any given agent, the quantization uncertainty is smaller than the GP prediction uncertainty following the expression in (3.4). Iteration k' is the last one in which the constraint is met. Therefore, in iteration $k' + 1$ we are forced to stop the algorithm because the threshold $\psi_i^{k'+1}$ will be smaller than our uncertainty measure that cannot decrease any further. However, at iteration k' we still satisfy the condition so the results of Theorem 7 hold leading to

$$\mathbb{E}[\epsilon^{k'+1}|\mathcal{I}^{k'}] \leq g\sqrt{\psi^{k'}} + \frac{1}{\sqrt{1+\delta}}\mathbb{E}[\epsilon^{k'}|\mathcal{I}^{k'-1}].$$

Then, we can do the same analysis as in the proof of Theorem 8 leading to the inequality

$$\mathbb{E}[\varepsilon^{k'+1}|\mathcal{I}^{k'}] \leq (\varepsilon^0 + gk')(r)^{k'}.$$

Figure 3.1 shows the plot of the bound $(\varepsilon^0 + gk)(r)^k$ where we can see that it increments up to a certain iteration at first and then starts decreasing indefinitely. The region where the function $d(k)$ decreases is given by

$$k \geq \frac{r}{(1-r)} - \frac{\varepsilon^0}{g}.$$

The constant $\frac{r}{(1-r)} - \frac{\varepsilon^0}{g}$ defines the iteration where $d(k)$ starts to decrease.

Unfortunately, having a bound for k' involves having a bound on trace$(\Sigma_i^{k'}(z_i^{k'}))$ that depends on k'. However, the error bound is not infinitely large even in the worst-case scenario, meaning that the uncertainty is always bounded. Considering that iteration k' is an extreme case and assuming a well-designed threshold mechanism, we anticipate that by the time we reach this moment the residual limit is not significantly large, as shown in Figure 3.1, so the solution we have at that moment is in the vicinity of the true solution. In the hypothetical case presented in Figure 3.1, considering that k' happens at iteration 220 then the bound on the residual is small at 0.027. This is mostly because our algorithms make sure that the overall uncertainty keeps decreasing at each iteration.

3.7. Discussion on Convergence Behavior of the Specific Approach presented in Chapter 2

In the previous subsections, we presented a convergence analysis for the STEP-GP and LGP algorithms when the query mechanism is performed by comparing the trace of the covariance matrix $\Sigma_i^k(z_i^k)$ to a decaying threshold instead of the maximum element of the diagonal of $\Sigma_i^k(z_i^k)$ as presented in Section 2.5.3.4. The following lemma presents a relationship between the convergence proof presented in Section 3.4 and the STEP-GP algorithm under the querying mechanism in Section 2.5.3.4.

Lemma 5 *Under the querying mechanism presented in Section 2.5.3.4, the STEP-GP*

algorithm converges and does so at a geometric rate.

Proof: The querying mechanism presented in Section 2.5.3.4 determines if communication is required following

$$\gamma_i^k = \begin{cases} 0, & \text{if } \max(\text{diag}(\Sigma_i^k(z_i^k))) \leq \psi_i^k \\ 1, & \text{otherwise,} \end{cases} \tag{3.9}$$

with local threshold $\psi_i^k = \psi_i^{k_0}(\alpha)^{k-k_0}$.

Since the trace of $\Sigma_i^k(z_i^k)$ is the sum of its diagonal entries, we can establish the following relationship

$$\text{trace}(\Sigma_i^k(z_i^k)) \leq p \max(\text{diag}(\Sigma_i^k(z_i^k))) \leq p\psi_i^k.$$

Assuming that the assessment to determine γ_i^k for each agent was already made, we take the sum over all agents:

$$\sum_{i=1}^{n} \left[(1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k))\right] \leq p \sum_{i=1}^{n} \left[(1-\gamma_i^k)\max(\text{diag}(\Sigma_i^k(z_i^k)))\right] \leq p \sum_{i=1}^{n} \psi_i^k.$$

The bound imposed on $\sum_{i=1}^{n} \left[(1-\gamma_i^k)\text{trace}(\Sigma_i^k(z_i^k))\right]$ (the same term used in Section 3.4) follows the same form of a constant multiplied by a geometrically decaying term. Since the sum of the maximum variances is bounded by this threshold form, Theorems 5 and 6 also apply to the querying mechanism presented in Section 2.5.3.4. This communication strategy imposes a tighter bound than that using the trace. □

On the other hand, the previous subsection presents a convergence analysis for the LGP algorithm when the quantization resolution is bounded. Theorems 7 and 8 show the convergence of the LGP algorithm using trace for the communication decision when the coordinator can vary the quantization resolution at each iteration and there is no bound

on the value that such resolution can take. The LGP algorithm presented in Chapter 2 has the added complication that its quantization resolution is fixed. For the case when the quantization resolution is bounded, we present a discussion in Section 3.6 where convergence is not concluded but it is shown that the expectation of the ADMM residual is bounded by a decaying bound. We are currently working on the convergence analysis when quantization is present, the quantization resolution is fixed, and the querying method presented in Section 2.5.3.4 is used. Those results will be presented in a future work. However, the empirical evidence of the extensive experiments performed suggests that the LGP algorithm as presented in Chapter 2 converges to an acceptable solution while not dramatically increasing the number of iterations required to reach convergence.

3.8. Conclusion to Chapter 3

In this chapter, we present a convergence analysis for the STEP-GP and LGP algorithms. The proofs were based on the convergence analysis of the generalized ADMM and SI-ADMM algorithms. For the case of the analysis of the LGP algorithm, we assumed that the coordinator can vary the quantization resolution at each iteration and that it can assign infinitely large bits for quantization. We also present convergence properties in the case where the quantization resolution is upper bounded using the LGP algorithm, leading to the conclusion that the expectation of the ADMM residual is bounded and such bound was explicitly stated. Finally, we present a connection between the analysis in this chapter and the algorithms defined in Chapter 2.

Chapter 4. Optimal Query Strategies for Communication-efficient ADMM using Gaussian Process Regression

Chapter 2 presented a hybrid approach combining an LMMSE regression with quantization to reduce the overall communication overhead when solving a distributed optimization problem with the ADMM algorithm. As explained in Section 2.5.3.4, the decision of whether communication is required at every iteration is done using a heuristic criterion utilizing the predictive covariance matrix of $\nabla f_i^{1/\rho}(z_i^k)$. Although the proposed hybrid approach resulted in a significant communication reduction, we believe that a refinement of the communication criterion could significantly impact the performance of ADMM using GP regression. Since this decision method directly affects how regression impacts the ADMM algorithm, we want to focus on ADMM performance when there is no quantization involved. Therefore, we could observe the impact of different query strategies on ADMM without being affected by the potential impact of the quantization error.

Chapter Organization: We begin with the problem formulation in Section 4.1. The systematic querying framework is presented in Section 4.2. We present our proposed joint query mechanism in Section 4.3, followed by our proposed individual query strategies in Section 4.4. A probabilistic comparison between the proposed methods, which leads to an expected querying behavior, is presented in Section 4.5. The numerical results are presented in Section 4.6, and the conclusions are made in Section 4.7.

This chapter previously appeared as: A. Duarte, T. X. Nghiem, S. Wei, Optimal Querying for Communication-efficient ADMM using Gaussian Process Regression, Franklin Open 6 (2024) 100080.

4.1. Problem Formulation

This chapter considers a sharing problem with n agents and a central coordinator, similar to that in [8, 10]. In this problem, a global cost, which includes all agents' strongly convex local cost functions f_i: $\mathbb{R}^p \mapsto \mathbb{R}$ on local decision variables $x_i \in \mathbb{R}^p$ and a convex shared cost function h: $\mathbb{R}^p \mapsto \mathbb{R}$, is minimized, as denoted by the expression

$$\text{minimize} \quad \sum_{i=1}^{n} f_i(x_i) + h\left(\sum_{i=1}^{n} x_i\right). \tag{4.1}$$

The cost function f_i is known only to its corresponding agent. Additionally, the problem (4.1) is solved with communication allowed only between the coordinator and agents, but without exchange between agents.

The sharing problem (4.1) is solved using ADMM as shown in [8] with the following updates

$$\begin{aligned} x_i^{k+1} &= \underset{x_i \in \mathbb{R}^p}{\arg\min} \left\{ f_i(x_i) + (\rho/2)\|x_i - x_i^k - \bar{y}^k + \bar{x}^k + u^k\|^2 \right\} \\ \bar{y}^{k+1} &= \underset{\bar{y} \in \mathbb{R}^p}{\arg\min} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^{k+1} - u^k\|^2 \right\} \\ u^{k+1} &= u^k + \bar{x}^{k+1} - \bar{y}^{k+1}, \end{aligned} \tag{4.2}$$

where k is the algorithmic iteration count, $\rho > 0$ is a penalty parameter and $\bar{x}^k = (1/n)\sum_{i=1}^{n} x_i^k$. In iteration k, the coordinator sends a query value z_i^k to the i-th agent and receives the following local *proximal operator* as a response

$$\mathbf{prox}_{\frac{1}{\rho} f_i}(z_i^k) = \underset{x_i \in \mathbb{R}^p}{\arg\min} \left\{ f_i(x_i) + \frac{\rho}{2}\|x_i - z_i^k\|^2 \right\}. \tag{4.3}$$

The x-minimization step in (4.2) consists of the local proximal minimization problem, for

every agent i,

$$x_i^{k+1} = \mathbf{prox}_{\frac{1}{\rho}f_i}(\underbrace{x_i^k + \bar{y}^k - \bar{x}^k - u^k}_{z_i^k}).$$

4.1.1. STructural Estimation of Proximal operator with Gaussian Processes (STEP-GP) Overview

For brevity, we will drop the subscript i and the superscript k in the subsequent equations. The Moreau envelope of f is defined as

$$f^{\frac{1}{\rho}}(z) = \min_{x\in\mathbb{R}^n} \left\{ f(x) + \frac{\rho}{2}\|x - z\|^2 \right\}. \tag{4.4}$$

When f is a convex function, the Moreau envelope $f^{\frac{1}{\rho}}$ is *convex and differentiable with Lipschitz continuous gradient with constant* ρ. Furthermore, given that the unique solution to the proximal minimization $x^{\frac{1}{\rho}}(z) = \mathbf{prox}_{\frac{1}{\rho}f}(z)$ is [35, Proposition 5.1.7]

$$x^{\frac{1}{\rho}}(z) = z - \frac{1}{\rho}\nabla f^{\frac{1}{\rho}}(z), \tag{4.5}$$

the optimal solution of (4.3) only requires the gradient $\nabla f^{\frac{1}{\rho}}(z)$ to be reconstructed. In [23], we proposed using GP to learn the local proximal operators, based on the training sets from past data to predict $\nabla f^{\frac{1}{\rho}}(z)$, thus improving the STEP method in [22]. This approach is named STEP-GP.

In particular, in STEP-GP, the coordinator maintains a GP model, named proxGP, for each agent. Each GP model predicts the gradient $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ of each agent's Moreau envelope, which has a multivariate Gaussian distribution with conditional mean $\mathbb{E}\left[\nabla f_i^{\frac{1}{\rho}}(z_i^k)\right] = \mu_i^k(z_i^k)$ and conditional covariance matrix $\mathrm{Cov}\left[\nabla f_i^{\frac{1}{\rho}}(z_i^k)\right] = \Sigma_i^k(z_i^k)$. The coordinator then uses an uncertainty measurement coming from the conditional covariance

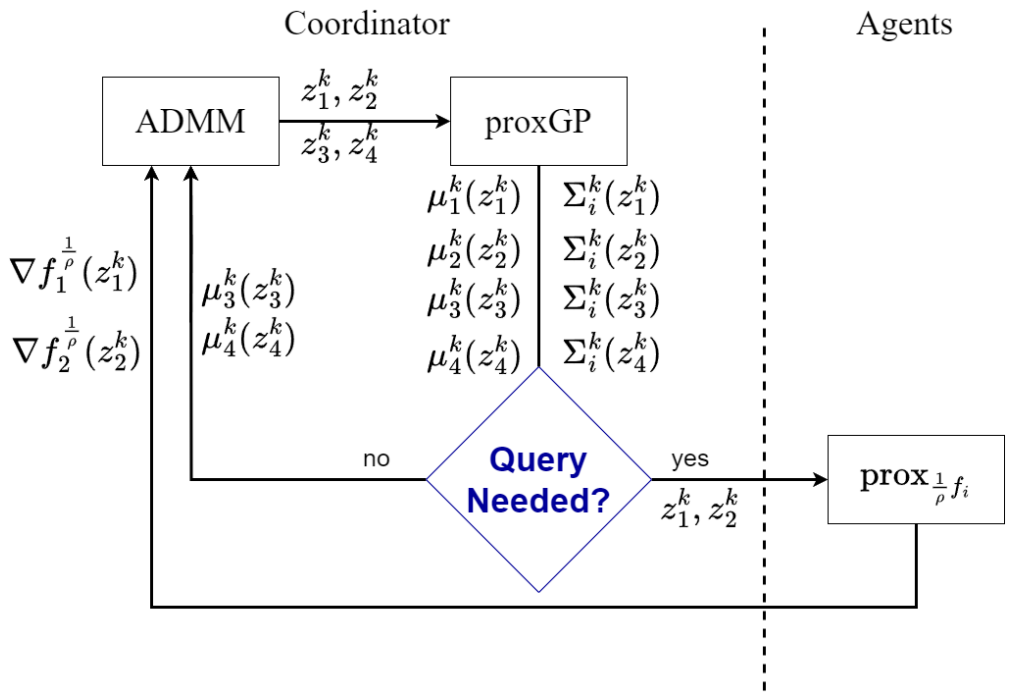

Figure 4.1. Flow diagram of the query decision and the query process and response between the coordinator and 4 agents in the proposed approach.

matrix to decide whether to query each agent. More details of the STEP-GP method can be found in [23].

4.1.2. Query-Response Dynamics

In Figure 4.1, we present one round of the proposed algorithm for a network of 4 agents. The GP regression block named proxGP refers to the GP prediction of $f_i^{\frac{1}{\rho}}(z_i^k)$ and $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ as presented in [23]. The coordinator has a corresponding proxGP for each agent, which is trained on its past query data with the agent. The coordinator first calculates the query variables z_i^k for each agent and uses them as input to the agent's proxGP. Using the covariance matrices $\Sigma_i^k(z_i^k)$ given by the proxGPs, the coordinator decides which agents are to be queried. In the figure, agents 1 and 2 are set to be queried, so the coordinator sends z_1^k and z_2^k to the agents, which solve their proximal minimization problems, depicted

by block $\mathbf{prox}_{\frac{1}{\rho}f_i}$. It then receives the Moreau envelopes $f_1^{\frac{1}{\rho}}(z_1^k)$, $f_2^{\frac{1}{\rho}}(z_2^k)$ and their gradients $\nabla f_1^{\frac{1}{\rho}}(z_1^k)$, $\nabla f_2^{\frac{1}{\rho}}(z_2^k)$ as responses from agents 1 and 2. Meanwhile, for agents 3 and 4, which are not queried, the coordinator uses the corresponding predicted values $\mu_3^k(z_3^k)$ and $\mu_4^k(z_4^k)$ from their proxGPs to perform the ADMM updates.

4.1.3. ADMM Updates with GP

Following the query-response mechanism presented in Figure 4.1, the ADMM expressions in (4.2) are modified to include the proxGP regression. First, let us define the communication decision variable for agent i at iteration k as

$$\gamma_i^k = \begin{cases} 1, & \text{if agent } i \text{ is queried} \\ 0, & \text{otherwise.} \end{cases} \tag{4.6}$$

When $\gamma_i^k = 1$, the query z_i^k is sent to agent i to obtain the exact value of $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$. On the contrary, when $\gamma_i^k = 0$, we use the predicted value $\mu_i^k(z_i^k)$ given by the GP. We then define the received value β_i^k as

$$\beta_i^k = \gamma_i^k \nabla f_i^{\frac{1}{\rho}}(z_i^k) + (1 - \gamma_i^k)\mu_i^k(z_i^k). \tag{4.7}$$

The ADMM expressions in (4.2) can now be reformulated as:

$$\begin{aligned} x_i^{k+1} &= z_i^k - (1/\rho)\beta_i^k \\ \bar{y}^{k+1} &= \underset{\bar{y}\in\mathbb{R}^p}{\arg\min} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^{k+1} - u^k\|^2 \right\} \\ u^{k+1} &= u^k + \bar{x}^{k+1} - \bar{y}^{k+1}. \end{aligned} \tag{4.8}$$

This chapter focuses on how the query decision-making, represented by the blue diamond block "Query Needed?" in Figure 4.1, can be carried out effectively.

4.2. General Querying Decision Framework

The main objective of including GP regression in the ADMM algorithm when solving a distributed optimization problem is to reduce communication overhead. However, we do not want it to significantly affect the algorithm convergence and accuracy of the optimization solution. A key component in the ADMM updates when GP is used, as presented in (4.8), is the variable β_i^k. This variable becomes the exact gradient $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ of the Moreau envelope or its predicted value, depending on γ_i^k. In (4.8), the set of x^{k+1}, $\bar{y}^k$, and u^{k+1} can be considered as a high dimensional vector trajectory to the global solution. This trajectory is affected by β_i^k, which depends on the communication decision variable γ_i^k as defined in (4.6) and (4.7), which in turn affects the GP regression accuracy and the optimization performance. Therefore, the mechanism to decide γ_i^k will ultimately impact the overall communication and optimization performance. If the coordinator does not have a sound and systematic mechanism to determine when to send queries to the agents, the ADMM algorithm could require excessive iterations to converge or never achieve convergence. Furthermore, it may reach an inaccurate solution upon reaching convergence. We propose a systematic querying framework that balances two opposing criteria: communication overhead reduction and optimization performance. In this framework, the querying decision solves an optimization of the form

$$\begin{aligned} \text{minimize} \quad & \text{comm}(\gamma^k), \\ \text{subject to} \quad & \gamma_i^k \in \{0,1\},\ 1 \leq i \leq n \\ & \text{uncer}(\gamma^k) \leq \psi^k, \end{aligned} \tag{4.9}$$

where $\text{comm}(\gamma^k)$ is a communication cost function, $\text{uncer}(\gamma^k)$ is an uncertainty function caused by the GP regression, and ψ^k is a given threshold that fluctuates at each iteration. The uncertainty is compared with the threshold because we want to limit the prediction error at each step so that the reduction in communication does not introduce an insurmountable amount of error to the ADMM algorithm. Therefore, the decision outcomes depend on how we measure those criteria. We can define the communication cost in several ways, such as the number of agents communicating at each iteration or the number of bits exchanged at each communication round. The uncertainty is measured by the prediction uncertainty of the agents' proxGPs. Thus, we define the query strategy in (4.9) as minimizing the communication cost under a constraint on the uncertainty introduced by proxGPs.

In general, the optimization problem (4.9) has to be solved using a combinatorial approach due to the n binary variables $\{\gamma_i^k\}_{i=1,\cdots,n}$. The computation cost, therefore, could be prohibitive when the number of agents is large. For that reason, in this chapter, we will seek approaches for solving (4.9) under certain communication cost and uncertainty functions without resorting to combinatorial techniques.

4.3. Proposed Joint Query Method

In this section, we propose a joint query strategy within the general framework, where the uncertainty function in (4.9) is the trace of the joint covariance matrix of the ADMM variables affected by the GP regression. In the following subsection, we justify why this uncertainty function is a suitable representation of the overall prediction error.

4.3.1. Justification of Adopting Trace of the Covariance Matrix as the Uncertainty Function

Consider a real Gaussian random vector $F \sim \mathcal{N}(\mu, \Sigma)$ with mean vector μ and covariance matrix Σ_F, where the l^{th} element of μ is μ_l, and the l^{th} element of F is F_l, with $l \in \{1, \ldots, p\}$. Our objective is to determine a sufficient condition for the L2 norm of the discrepancy between F and its mean to be small with high probability. This can be expressed by the confidence sphere:

$$\mathrm{P}\big[\|F - \mu\|_2 \leq \|\mu\|_2 \delta\big] \geq 1 - \xi, \tag{4.10}$$

where ξ and δ are two small numbers chosen in advance for quality control. The values of δ and ξ must be small because we want the discrepancy between the actual value and the mean of F to be small with high probability, so these control variables will determine how tight we allow the discrepancy to be and with how much probability.

The following proposition presents a sufficient condition for (4.10).

Proposition 4 *A sufficient condition for* (4.10) *is given by*

$$\mathrm{tr}(\Sigma_F) \leq \|\mu\|_2^2 \delta^2 - 2\left(\lambda_1 \ln(1/\xi) + \sqrt{\ln(1/\xi)}\sqrt{\sum_{l=1}^{p} \lambda_l^2}\right). \tag{4.11}$$

Proof: The proof is presented in Appendix G. □

Proposition 4 suggests that the trace of the joint covariance matrix of the ADMM variables affected by GP regression, as the random vector F, can be constrained to control the desired prediction errors, which affect the convergence of the algorithm and the accuracy of the solution. Therefore, it justifies the use of this trace as the uncertainty function $\mathrm{uncer}(\gamma^k)$ in (4.9).

4.3.2. Proposed Joint Query Method

Following the general querying decision framework presented in Section 4.2, we propose using the L1 norm of γ^k as the communication cost function, which indicates how many agents are queried in the current iteration.

The uncertainty function uncer$(\cdot)$ is selected based on the analysis in the previous subsection and the work [46]. The authors of [46] present a stochastic approach to inexact ADMM in which the expectation of the mean square error of the inexact ADMM variables with respect to their exact counterparts is bounded. It can be shown that the bounded expectation is equal to the trace of the error covariance matrix. Extending both analyses to our problem, we propose to use the trace of the joint covariance matrix of the iterative variables of the ADMM algorithm, given by $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}|\gamma^k])$, to derive the uncertainty function. Here, $\text{tr}(\cdot)$ is the trace operator.

We thus have the following realization of the general optimization problem (4.9):

$$\begin{aligned} &\text{minimize} && \|\gamma^k\|_1 \\ &\text{subject to} && \gamma_i^k \in \{0,1\}, 1 \leq i \leq n, \\ & && \text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}]|\gamma^k]) \leq \psi^k, \end{aligned} \tag{4.12}$$

where the threshold ψ^k varies at each iteration. The rationale for (4.12) is to choose the smallest set of agents to query while ensuring that the trace of the joint uncertainty caused by not querying the other agents does not exceed the threshold ψ^k, thus ensuring that there is a high probability that the uncertainty is within a desired sphere. Following the convergence analysis for the stochastic inexact ADMM in [46], we choose the sequence of thresholds ψ^k such that $\sum_{k=1}^{\infty} \psi^k < \infty$. More details on ψ^k are presented in

Section 4.3.4.

Next, we present an efficient solution to the problem in (4.12) without resorting to a combinatorial approach by exploiting the convexity and linearity of the cost functions and constraints considered. The idea is that the search for a set of agents to query starts with the scenario where the communication cost is maximum and the uncertainty is minimum. Then, we calculate the contribution to the joint trace of each agent where the ones that contribute the least to the joint uncertainty will be the first candidates not to be queried in the current round. Instead of considering each possible combination, we analyze the constraint on the joint uncertainty each time the next candidate is set to skip communication until the constraint is met. The proposed joint query method named *L1Norm-Trace* follows the steps listed below at iteration k:

1. For each agent, calculate its uncertainty contribution $\text{un}_i = \text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1} | \gamma_i^k = 0, \gamma_{j\neq i}^k = 1])$.

2. In the order from the smallest to the largest un_i, pick all the agents whose sum of un_i does not exceed the threshold ψ^k and set their γ_i^k to 0, i.e., they are not queried. The remaining agents are to be queried, i.e., their γ_i^k are set to 1.

The proposed strategy does not consider all possible combinations of communicating agents, as it would be necessary to combinatorically solve the problem posed in (4.12). However, our strategy solves this optimization problem optimally.

Lemma 6 *The L1Norm-Trace method solves the optimization problem in* (4.12) *optimally.*

Proof: If our method is not optimal, then our selection of agents to be queried does not minimize the communication cost while ensuring that the uncertainty constraint is met. Because we select agents from the smallest to the largest un_i, we select the

largest number of agents to not be queried such that the uncertainty constraint is met. There is no other selection of agents that can further reduce $\|\gamma^k\|_1$ without violating $\sum_{i=1}^{n} \text{tr}(\text{Cov}[x_i^{k+1}; \bar{y}_i^{k+1}; u_i^{k+1}]|\gamma_i^k]) \leq \psi^k$. □

The next subsections derive the calculation of the joint trace $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}]|\gamma^k])$ and present the mechanism to vary the threshold ψ^k.

4.3.3. Derivation of the Trace of the ADMM Joint Covariance Matrix

In this subsection, we first present an equivalent expression to the ADMM updates presented in (4.8) that allows us to see the inherent coupling of the agents. This expression is then used to find the specifics of the proposed uncertainty cost $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}]|\gamma^k])$. The following proposition uses the notation presented in the problem definition in Section 4.1.

Proposition 5 *The specific form of the ADMM algorithm presented in* (4.8) *has an equivalent expression given by*

$$\begin{aligned} x_i^{k+1} &= z_i^k - (1/\rho)\beta_i^k \\ u^{k+1} &= (1/\rho)\nabla h^{n/\rho}\left(v^k\right) \\ \bar{y}^{k+1} &= \bar{y}^k - 1/(\rho n)\sum_{i=1}^{n} \beta_i^k - u^{k+1}, \end{aligned} \tag{4.13}$$

where $v^k = n\bar{y}^k - (1/\rho)\sum_{i=1}^{n} \beta_i^k$ *and* $\nabla h^{n/\rho}()$ *is the gradient of the Moreau Envelope of the function* h.

Proof: The proof is presented in Appendix H. □

The expression in (4.13) presents the ADMM updates in terms of the gradient of the Moreau Envelope of functions $\{f_i\}$ and h, and follows the calculations for the ADMM algorithm executed on the coordinator side. More importantly, such an expression also

shows that each agent's β_i^k is present in each of the ADMM updates, especially in the $\bar{y}^{k+1}$ and u^{k+1} updates where we have the sum of those variables. The variable β_i^k (depending on γ_i^k as defined in (4.7)) comes from the exact value or the predicted value of $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$, so the ADMM updates in (4.13) can be used to quantify the joint uncertainty of the ADMM variables.

Due to the linearity of the trace, the proposed uncertainty cost is simplified to $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}|\gamma^k]) = \text{tr}(\text{Cov}[x^{k+1}|\gamma^k]) + \text{tr}(\text{Cov}[\bar{y}^{k+1}|\gamma^k]) + \text{tr}(\text{Cov}[u^{k+1}|\gamma^k])$. Following the expression in (4.13), the definition of β_i^k in (4.7), and that only the terms including β_i^k contribute to the uncertainty, the overall trace function becomes

$$\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}|\gamma^k]) = (1 + 1/n^2)(1/\rho)^2 \sum_{i=1}^{n} (1 - \gamma_i^k) \,\text{tr}\left(\Sigma_i^k(z_i^k)\right) + 2(1/\rho)^2 \,\text{tr}\left(\text{Cov}[\nabla h^{n/\rho}(v^k)]\right), \quad (4.14)$$

which is subject to the function h. Calculating the covariance matrix of $\nabla h^{n/\rho}(v^k)$ given the probability distribution of v^k is generally difficult and may not have a closed-form equation, because $\nabla h^{n/\rho}(\cdot)$ is generally a nonlinear function. In this case, we must approximate this covariance matrix [47]. However, this approximation will introduce uncertainty, which will propagate into the algorithmic iterations, affecting the communication decision methods and having an impact on the ADMM algorithm.

4.3.4. Threshold ψ^k Mechanism

During the execution of the ADMM algorithm, the uncertainty of the GP regression tends to reduce when the ADMM algorithm gets closer to convergence. This is because more training data from responses to queries is available, which allows the prediction to be more accurate. For that reason, the threshold to be considered should decrease

over the ADMM iterations. We propose a decreasing threshold mechanism that relies on the iteration count and k_0, which is the iteration where the GP regression is used for the first time.

$$\psi^{k_0} = \iota V^{k_0}, \tag{4.15}$$

where ι, chosen in advance, is a number between 0 and 1, and V^{k_0} is the uncertainty variable used by the query method (in this case $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}]|\gamma^k])$). Given a preselected decay rate $\alpha \in (0, 1)$, at a later iteration $k > k_0$, the threshold is updated as:

$$\psi^k = \psi^{k_0} \alpha^{k-k_0}. \tag{4.16}$$

4.4. Proposed Individual Query Methods

In this section, we simplify the query framework presented in Section 4.2 by proposing three individual query methods to determine when a communication round between the coordinator and the agents is necessary. The notation individual query method is used to describe that the coordinator determines if communication with a specific agent is required by analyzing its uncertainty individually without considering the uncertainty measures of the other agents. This strategy reduces considerably the computational complexity of the general method presented in Section 4.2, but ignores the impact of an agent's decision on the overall prediction error introduced to the system. However, by limiting the uncertainty of each agent per iteration, we ensure that the prediction error does not affect the ADMM's algorithm performance greatly. Although this approach is not as rigorous as the joint method, its simplicity makes it suitable for applications where the computational cost must be as low as possible.

In an individual query method, the decision is made per agent where this decision

is reflected in the agent's corresponding binary decision variable γ_i^k. The general principle of such methods is that for agent i, the coordinator shall decide in favor of not sending a query to this agent if the probability of an estimation error of both the Moreau Envelope and its gradients is within an acceptable bound. This estimation error is quantified in different ways. By doing this, we drop the minimization problem presented in (4.9) and set each γ_i^k by comparing the estimated error of each agent to a threshold individually. The individual query strategies proposed were not arbitrarily derived, but followed the mathematical intuition given by a confidence interval analysis to be performed per agent. The specifics of the proposed individual query strategies are presented in the following subsections.

4.4.1. Maximum Variance Query Method

Similarly to the derivation presented in Section 4.3.1, our goal is to generate a decision rule in which the prediction error is small with a high probability. For that reason, using the concept of confidence interval, a threshold setting can be derived. When the prediction error is below a chosen threshold, no query will be sent to an agent. As a consequence, we want the probability that the estimation error is bounded by a small upper bound to be as large as possible.

For the following derivations, we employ the general notation used in Section 4.3.1, where the variables F, F_l, μ, μ_l, δ, and ξ were defined, and we add the definition of the vector of variances of F as $s^2 = \text{diag}(\Sigma_F)$, where the l^{th} element of s^2 is s_l^2. The desired confidence interval is given by

$$\text{P}\left[-\delta\|\mu\|_1 \leq \|F-\mu\|_1 = \sum_{l=1}^{p} |F_l - \mu_l| \leq \delta\|\mu\|_1\right] \geq 1-\xi, \tag{4.17}$$

A sufficient condition of (4.17) is given below in terms of the requirement imposed on each dimension F_l of F.

$$\mathrm{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}, 1 \leq l \leq p\right] \geq 1-\xi. \tag{4.18}$$

Following the region probability defined in [48], we get an immediate bound of (4.18):

$$\mathrm{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}, 1 \leq l \leq p\right] \geq \prod_{l=1}^{p} \mathrm{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}\right], \tag{4.19}$$

and it implies that if the following condition holds true,

$$\mathrm{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}\right] \geq 1-\xi', \forall 1 \leq l \leq p, \tag{4.20}$$

where $1-\xi' = (1-\xi)^{1/p}$, the requirement in (4.17) is immediately satisfied.

However, instead of analyzing this condition for each of the dimensions of F, we can simplify the analysis by further requiring that the maximum standard deviation (the maximum element of the vector s) satisfy the condition inside the probability in (4.18) when the bound is minimum. This is achieved when

$$\mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l} \leq \frac{\delta \min_{1\leq l\leq p}|\mu_l|}{\max_{1\leq l\leq p}(s_l)}\right] \geq 1-\xi', \forall 1 \leq l \leq p, \tag{4.21}$$

The condition in (4.20) is met when requiring

$$\max_{1\leq l\leq p}(s_l) \leq \frac{\min_{1\leq l\leq p}|\mu_l|\delta}{Q^{-1}(\xi'/2)} = \psi^{(1)}, \tag{4.22}$$

where $Q^{-1}()$ is the inverse of the Q-function $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} e^{-v^2/2} dv$. The right-hand side of the inequality in (4.22) can be used as the threshold $\psi^{(1)}$ to compare the maximum element of the vector of variances s (s_l). In case $\max_{1\leq l\leq p}(s_l) \leq \psi^{(1)}$, then automatically all the elements of s satisfy the condition.

In the context of the problem defined in Section 4.1, at each iteration, the GP regression gives us for agent i the predicted mean $\mu_i^k(z_i^k)$ and the conditional covariance matrix $\Sigma_i^k(z_i^k)$. In this scenario, the vector of variances will be defined as $(s_i^k)^2 = \text{diag}(\Sigma_i^k(z_i^k))$. Furthermore, as mentioned in the previous section, each agent's GP prediction uncertainty is reduced over the algorithmic rounds. For that reason, the threshold $\psi^{(1)}$ should not be static as also implied in (4.22) but should decrease over the ADMM iterations. This requires the control variables ξ and δ to be adjusted at each iteration, which can be problematic considering that the two variables need to be adjusted at each round. Therefore, we do not use the specific threshold $\psi^{(1)}$ defined in (4.22), but instead employ a general threshold ψ_i^k per agent that follows the threshold mechanism described in Section 4.3.4. Finally, under this querying mechanism, the variable γ_i^k is defined as

$$\gamma_i^k = \begin{cases} 0, & \text{if } \max_{1\leq l\leq p}(s_{i[l]}^k) \leq \psi_i^k \\ 1, & \text{otherwise.} \end{cases} \tag{4.23}$$

4.4.2. Maximum Variance and Mean Ratio Query Method

The subsequent proposed strategy expands from the confidence interval analysis presented in Section 4.4.1 to build its mathematical intuition. Following the confidence interval defined in (4.18), to require that each dimension of an agent has a small relative estimation error, we are interested in evaluating the bound in (4.19). Defining $a^* = \max_{1\leq l\leq p}\frac{s_l}{|\mu_l|}$, it is then straightforward to show that if

$$\prod_{l=1}^{p} \text{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}\right] \geq \left(\text{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta}{a^*}\right]\right)^p \geq 1-\xi, \tag{4.24}$$

we always satisfy

$$\mathrm{P}\left[\left|\frac{F_l-\mu_l}{s_l}\right| \leq \frac{\delta|\mu_l|}{s_l}, 1 \leq l \leq p\right] \geq 1-\xi. \tag{4.25}$$

Note that under the GP model, each F_l is Gaussian following $\mathcal{N}(\mu_l, s_l^2)$, suggesting $\frac{F_l-\mu_l}{s_l}$ following $\mathcal{N}(0,1)$. We then obtain a sufficient condition to meet the confidence region requirement stated in (4.25), namely,

$$\max_{1\leq l\leq p}\frac{s_{[l]}}{|\mu_{[l]}|} \leq \frac{\delta}{Q^{-1}(1/2-1/2*(1-\xi)^{1/p})} = \psi^{(2)}. \tag{4.26}$$

The upper-bound expressed in (4.26) is not imposed on the maximum element of s but on the maximum ratio of $\frac{s_l}{|\mu_l|}$.

In the context of our problem defined in Section 4.1, the threshold $\psi^{(2)}$ should decrease over the ADMM algorithmic rounds to keep up with the reduction of the uncertainty of the GP prediction. Similarly to the query method presented in Section 4.4.1, we do not use the specific threshold $\psi^{(2)}$ defined in (4.26), but instead employ a general threshold ψ_i^k per agent following the mechanism described in Section 4.3.4. Using the notation of our problem, the variable γ_i^k under this query strategy is defined as

$$\gamma_i^k = \begin{cases} 0, & \text{if } \max\limits_{1\leq l\leq p}\frac{s^k_{i[l]}}{|\mu^k_{i[l]}(z_i^k)|} \leq \psi_i^k \\ 1, & \text{otherwise.} \end{cases} \tag{4.27}$$

4.4.3. Ratio of Maximum Eigenvalue and the Norm of the Mean Query Method

In this subsection, we derive a norm-based decision strategy about when a query shall be sent to an agent by the coordinator similar to the one derived in Section 4.3.4. Our objective is to fulfill the decision criterion presented in (4.10) given by:

$$\mathrm{P}\big[\|F-\mu\|_2 \leq \|\mu\|_2\delta\big] \geq 1-\xi.$$

Following the same transformation presented in Appendix G expressed in (G.1), we seek an alternative sufficient condition to satisfy the confidence sphere condition in (G.1). We find an alternative lower bound on this probability by defining $\lambda_1 = \max_{1\leq l\leq p} \lambda_l$ (the maximum eigenvalue of the matrix Σ_F) and resorting to the following inequality

$$\sum_{l=1}^{p} \frac{G_l^2}{\lambda_l} \geq \frac{1}{\lambda_1} \sum_{i=1}^{p} |G_l|^2 = \frac{1}{\lambda_1} \|G\|^2, \tag{4.28}$$

where $G_l/\sqrt{\lambda_l}$ are independent and identical distributed (i.i.d standard Gaussian following $\mathcal{N}(0,1)$, which suggests that $\sum_{l=1}^{p} \frac{G_l^2}{\lambda_l}$ follows a chi-square distribution with degree of p, i.e. $\sum_{l=1}^{p} \frac{G_l^2}{\lambda_l} \sim \chi_p^2$. Based on the desired bound in (4.10) and the inequality in (G.1), we have a sufficient condition to satisfy (4.10) given by:

$$\mathrm{P}\big[\|G\|_2 \leq \|\mu\|_2 \delta\big] \geq \mathrm{P}\left[\sum_{l=1}^{p} \frac{G_l^2}{\lambda_l} \leq \frac{1}{\lambda_1} \|\mu\|_2^2 \delta^2\right] \geq 1-\xi. \tag{4.29}$$

This expression can be satisfied if λ_1 satisfies the following condition:

$$\frac{\lambda_1}{\|\mu\|_2^2} \leq \frac{\delta^2}{\mathcal{F}_{\chi^2}^{-1}(1-\xi)} = \psi^{(3)}, \tag{4.30}$$

where $\mathcal{F}_{\chi^2}^{-1}(.)$ is the inverse function of the Cumulative Distribution Function (CDF) of the chi-square random variable. Thus, if $\frac{\lambda_1}{\|\mu\|_2^2} \leq \psi^{(3)}$, we ensure that the confidence sphere criterion in (G.1) is met; therefore, there is no need to send a query. It should be noted that, different from the approach following a high-dimensional confidence region whose sufficient condition is based on the maximum ratio between the standard deviation and its associated absolute mean, as stated in (4.26), we need to compare the relationship between the maximum eigenvalue and the square of the L2 norm of the conditional mean to a threshold subject to the chi-square distribution, under the confidence sphere setting. Once again, the specific threshold presented in this subsection is replaced by a general

threshold ψ_i^k per agent following the mechanism described in Section 4.3.4. Finally, we define a query strategy in which the variable γ_i^k is defined as

$$\gamma_i^k = \begin{cases} 0, & \text{if } \frac{\lambda_1^k}{\|\mu_i^k(z_i^k)\|_2^2} \leq \psi_i^k \\ 1, & \text{otherwise.} \end{cases} \tag{4.31}$$

The query strategies presented in this section are simple strategies with low impact on the overall computational cost, but they ignore the inherent uncertainty dependencies between the agents which will negatively affect the performance of the ADMM algorithm. The following section presents a comparative analysis of the mathematical foundation of each of the proposed methods to have an intuition about what querying behavior to expect for each method.

4.5. Probability Comparison Between Querying Strategies

In this section, we present a comparative analysis of the probabilities used as a basis for the various querying strategies proposed. This analysis allows us to have an idea of the expected querying behavior for each of the methods. For the following derivations, we use the same notation used to derive each of the methods' probabilities first defined in Section 4.3.1.

4.5.1. Relationship between Maximum Variance and Maximum Ratio Methods

Comparing the conditions presented in (4.20) and (4.24), while acknowledging the bound presented in (4.19), we can observe that the condition in (4.20) is more likely to occur. Thus, we find that the relationship between the Maximum Variance and Maximum

Ratio between variance and mean methods is given by

$$\left(\mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l}\leq\frac{\delta}{a^*}\right]\right)^p\leq\mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l}\leq\frac{\delta\min_{1\leq l\leq p}|\mu_l|}{\max_{1\leq l\leq p}(s_l)},1\leq l\leq p\right]. \tag{4.32}$$

This relationship shows that the condition given by the maximum ratio method is more stringent than the one for the maximum variance. For that reason, we anticipate the former to behave more aggressively in terms of the frequency of queries.

4.5.2. Relationship between L2 Norm-based Methods and a L1 Norm condition

The querying strategies involving the maximum eigenvalue and the trace, presented in Sections 4.4.3 and 4.3.1, respectively, are derived from the same confidence sphere involving the L2 norm of $F-\mu$. This confidence region is defined in Equation (4.10). We want to find a relationship between this confidence sphere and a condition involving the L1 norm of $F-\mu$ given by

$$\mathrm{P}\left[\|F-\mu\|_1\leq\delta\|\mu\|_2\right]\geq 1-\xi. \tag{4.33}$$

We know that for any real vector x, the relationship between L1 and L2 norms is given by $\|x\|_1\geq\|x\|_2$. This implies

$$\mathrm{P}\left[\|F-\mu\|_1\leq\delta\|\mu\|_2\right]<\mathrm{P}\left[\|F-\mu\|_2\leq\delta\|\mu\|_2\right], \tag{4.34}$$

which suggests that if the condition in (4.33) holds true then automatically the condition in (4.10) is also true, thereby the querying condition based on L1 norm is more demanding than that under the L2 norm, thereby resulting more frequent queries accordingly.

4.5.3. Relationship between Maximum Variance Method and an L1 Norm condition

The probability in the condition given in (4.33) can be expressed as

$$\mathrm{P}\left[\sum_{l=1}^{p}|F_l-\mu_l|\leq\delta\|\mu\|_2\right]\geq 1-\xi. \tag{4.35}$$

Since a sufficient condition of $\sum_{l=1}^{p}|F_l-\mu_l|\leq\delta$ is $|F_l-\mu_l|\leq\frac{1}{p}\delta\|\mu\|_2$, for $1\leq l\leq p$, we have

$$\mathrm{P}\left[|F_l-\mu_l|\leq\frac{1}{p}\delta\|\mu\|_2,\ 1\leq l\leq p\right]\leq\mathrm{P}\left[\|F-\mu\|_1\leq\delta\|\mu\|_2\right]. \tag{4.36}$$

Now, we want to compare the left-hand side of (4.36) with the probability expression for the Maximum Variance method in (4.18). Since the variable δ, used throughout all derived probabilities, is a variable that can be tuned, we can define a variable $\hat{\delta}$ such that $\frac{1}{p}\hat{\delta}\|\mu\|_2=\delta\min_{1\leq l\leq p}|\mu_l|$. Dividing by s_l into both sides of the arguments in the probability of the left side of (4.36), it is straightforward to see that the following inequalities hold

$$\begin{aligned}\mathrm{P}\left[\frac{\left|F_{[l]}-\mu_{[l]}\right|}{s_l}\leq\frac{\delta\min_{1\leq l\leq p}|\mu_l|}{\max_{1\leq l\leq p}(s_l)},1\leq l\leq p\right]\leq\mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l}\leq\frac{1}{p}\frac{\hat{\delta}\|\mu\|_2}{s_l},\ 1\leq l\leq p\right]\\ \leq\mathrm{P}\left[\|F-\mu\|_1\leq\hat{\delta}\|\mu\|_2\right].\end{aligned} \tag{4.37}$$

This results in the condition based on the L1 norm of $F-\mu$ being more likely to occur than the condition used in the query method based on the maximum variance.

4.5.4. Complete Relationship Between all Methods

Combining the inequalities obtained in (4.32), (4.34), and (4.37) with the definition of $\hat{\delta}$, we get the following inequalities

$$\begin{aligned} & \left(\mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l} \leq \frac{\delta}{a^*}\right]\right)^p \\ \leq\ & \mathrm{P}\left[\frac{|F_l-\mu_l|}{s_l} \leq \frac{\delta \min_{1\leq l\leq p}|\mu_l|}{\max_{1\leq l\leq p}(s_l)}, 1 \leq l \leq p\right] \\ \leq\ & \mathrm{P}\left[\|F-\mu\|_1 \leq \hat{\delta}\|\mu\|_2\right] \\ \leq\ & \mathrm{P}\left[\|F-\mu\|_2 \leq \hat{\delta}\|\mu\|_2\right]. \end{aligned} \tag{4.38}$$

The relationships in (4.38) demonstrate how the probabilities used in our proposed decision strategies are related to each other. They reveal that the query dynamics will be more aggressive when using the method based on the maximum ratio of mean and variance, followed by the method based on the maximum variance, and finally, the two methods directly based on the L1 and L2 norm-based confidence spheres will end up with a more relaxed querying dynamics.

The following section presents numerical results to validate and compare all the proposed query methods. We will present comparisons made in terms of querying dynamics, which will be shown consistent with the analysis presented in this section and their resulting convergence speed and qualities in solving a distributed ADMM optimization problem.

4.6. Numerical Results

In this section, we evaluate the proposed query methods through a numerical study of solving a sharing problem where each agent's local function is quadratic.

The details of our problem setting are presented next.

4.6.1. Quadratic Sharing Problem

4.6.1.1. Problem Definition

We evaluate our methods using a sharing problem motivated by the application in [10]. However, we do not consider the dynamic behavior of the variables as in [10] but assume that they are stationary. The sharing problem is formulated as

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n}[(1/2)x_i^T M_i x_i + w_i^T x_i + c_i] \\ & + (1/2)\sum_{i=1}^{n} y_i^T M_h \sum_{j=1}^{n} y_j + w_h^T \sum_{i=1}^{n} y_i + c_h \\ \text{subject to} \quad & x_i - y_i = 0, \end{aligned} \tag{4.39}$$

where for $i = 1, \cdots, n$, variables $x_i, y_i \in \mathbb{R}^p$, with $w_i, w_h \in \mathbb{R}^p$, $M_i, M_h \in \mathbb{R}^{p\times p}$ positive definite, and $c_i, c_h \in \mathbb{R}$ being given problem parameters.

4.6.1.2. Problem Parameters Generation

The problem's parameters presented (4.39) are generated following the example given in [10]. First, the parameters c_i and c_h will be two randomly generated numbers that are uniformly distributed on [-1,1]. For the case of w_i, we generate for each agent a parameter $w_i^{[0]}$ which is a p-dimensional vector with entries randomly generated and uniformly distributed on [-1,1]. Then, the value of w_i is generated for each agent following $w_i = w_i^{[0]} + \eta s_i$, where η is some small positive number and s_i is a p-dimensional vector for agent i whose entries are randomly generated and uniformly distributed on [-1,1]. The parameter w_h is generated following the same procedure as w_i, but is calculated only once and not for each agent.

On the other hand, to generate M_i for each agent, we first generate a symmetric $p \times p$ matrix $M_i^{[0]} = AA'$, where the entries of A are randomly generated and uniformly

distributed on $[-1, 1]$. Then we generate $\tilde{M}_i = M_i^{[0]} + \eta S_i$, where $S_i = BB'$ is a symmetric $p \times p$ matrix with the entries of B randomly generated and uniformly distributed on [-1,1]. Subsequently, M_i is constructed as

$$M_i = \begin{cases} \tilde{M}_i, & \text{if } \lambda_{min}(\tilde{M}_i) > \epsilon_d \\ \tilde{M}_i + \left(\epsilon_d - \lambda_{min}(\tilde{M}_i)\right) I_p, & \text{otherwise,} \end{cases} \tag{4.40}$$

where $\lambda_{min}(\tilde{M}_i)$ denotes the smallest eigenvalue of $\tilde{M}_i$ and $\epsilon_d > 0$ is a positive constant. The parameter M_h is generated following the same procedure as M_i, but it is calculated only once and not for each agent.

4.6.1.3. Solution Using ADMM

Following the specifics of the problem in (4.39) and the expression of ADMM in (4.2), we can derive a closed-form solution for updating the ADMM variable x_i^{k+1}. Because the function f_i is convex, the optimal solution of x_i^{k+1} is attained when the gradient of the objective function vanishes. By taking the gradient of the x_i^{k+1}-update and equating it to zero, we obtain

$$x_i^{k+1} = (M_i + \rho I_p)^{-1}(\rho z_i^k - w_i), \tag{4.41}$$

where I_p is the $p \times p$ identity matrix. The expression in (4.41) is the closed-form solution of the optimization for the x_i update to be computed on the agent side.

Similarly, we can derive a closed-form solution for the $\bar{y}^{k+1}$ update. Because the function h is also convex quadratic then once again the optimal solution of $\bar{y}^{k+1}$ is attained when the gradient of the objective function vanishes, leading to the expression

$$\bar{y}^{k+1} = (nM_h + \rho I_p)^{-1}(\rho(u^k + \bar{x}^{k+1}) - w_h). \tag{4.42}$$

Finally, combining the ADMM expression in (4.2) with the expressions in (4.41) and (4.42), the ADMM updates are expressed as

$$\begin{aligned} x_i^{k+1} &= (M_i + \rho I_p)^{-1}(\rho z_i^k - w_i) \\ \bar{y}^{k+1} &= (nM_h + \rho I_p)^{-1}((\rho/n)v^k - w_h) \\ u^{k+1} &= (1/n)(v^k - n\bar{y}^{k+1}), \end{aligned} \tag{4.43}$$

where $v^k = n\bar{y}^k - (1/\rho)\sum_{i=1}^n \beta_i^k$.

4.6.2. Equation of the Trace of the Joint Covariance Matrix

As presented in Section 4.3.2, our proposed joint query strategy depends on an uncertainty measurement given by the trace of the joint uncertainty of the ADMM updates. The specific expression of $\text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}]|\gamma^k])$, following the specific ADMM updates presented in (4.43), is given by

$$\begin{aligned} \text{tr}(\text{Cov}[x^{k+1}; \bar{y}^{k+1}; u^{k+1}|\gamma^k]) = (1 + 1/n^2)(1/\rho)^2 \sum_{i=1}^n (1 - \gamma_i^k)\text{tr}\left(\Sigma_i^k(z_i^k)\right) + \\ (2/n^2)\sum_{i=1}^n (1 - \gamma_i^k)\text{tr}\left(C^T C \Sigma_i^k(z_i^k)\right) - 2(1/n^2\rho)\sum_{i=1}^n (1 - \gamma_i^k)\text{tr}\left(C\Sigma_i^k(z_i^k)\right), \end{aligned} \tag{4.44}$$

where $C = (nM_h + \rho I_p)^{-1}$.

4.6.3. Numerical Implementation

The problem in (4.39) is solved with two different algorithms:

1. *Sync*: this algorithm uses ADMM with proximal operator as in (4.2), which simplifies to (4.43) with $\rho = 10$.

2. *STEP-GP*: the algorithm proposed in [23].

For the *STEP-GP* algorithm, different query methods are considered as follows:

- *MaxVar*: The query strategy presented in Section 4.4.1.

- *MaxRat*: The query strategy presented in Section 4.4.2.
- *MaxEig*: The query strategy presented in Section 4.4.3.
- *L1Norm-Trace*: The query strategy presented in Section 4.3.2.

In our numerical computations, we consider the following combinations: *Sync*, *STEP-GP:MaxVar*, *STEP-GP:MaxRat*, *STEP-GP:MaxEig*, and *STEP-GP:L1Norm-Trace*. Also, we consider two cases where the number of agents is taken from $n \in \{10, 30\}$.

Our results were generated using MATLAB. For comparison purposes, ground truth solutions to minimization problems (4.39) were obtained using the YALMIP toolbox [41]. For the construction of the GP models, we used the GPstuff toolbox [42]. All calculations were performed on high-performance computers at Louisiana State University (http://www.hpc.lsu.edu).

4.6.4. Metrics and Considerations

4.6.4.1. Media Access Control (MAC) Metric

Appendix F presents the details of how this metric is obtained.

4.6.4.2. ADMM Termination Criterion

For our numerical computations, we use the ADMM termination criterion presented in Section 3.3.1 in [8]. This criterion presents two conditions that compare the primal and dual of ADMM against two different tolerances. Expressing the primal and dual in terms of the specifics of our problem results in a termination criterion of the form:

$$\|\bar{x}^{k+1} - \bar{y}^{k+1}\|_2 \leq \epsilon^{\text{pri}} \text{ and } \|\rho(\bar{y}^{k+1} - \bar{y}^{k})\|_2 \leq \epsilon^{\text{dual}}, \tag{4.45}$$

where $\epsilon^{\text{pri}} > 0$ and $\epsilon^{\text{dual}} > 0$ are feasibility tolerances for the primal and dual feasibility conditions. These tolerances can be chosen using an absolute and relative criteria, such as

$$\epsilon^{\text{pri}} = \sqrt{p}\epsilon^{\text{abs}} + \epsilon^{\text{rel}} \max(\|\bar{x}^{k+1}\|_2, \|\bar{y}^{k+1}\|_2),$$
$$\epsilon^{\text{dual}} = \sqrt{p}\epsilon^{\text{abs}} + \epsilon^{\text{rel}} \|\bar{y}^{k+1}\|_2,$$

where $\epsilon^{\text{abs}} > 0$ is an absolute tolerance, $\epsilon^{\text{rel}} > 0$ is a relative tolerance, and the factor $\sqrt{p}$ account for the fact that the L2 norms are in $\mathbb{R}^p$. Both ϵ^{abs} and ϵ^{rel} are set manually at the beginning of the algorithm. The choice of ϵ^{abs} depends on the scale of the typical variable values of the application, while reasonable values for ϵ^{rel} might be 10^{-3} or 10^{-4}, depending on the application.

4.6.4.3. Performance Trade-off

We propose to present the results showing directly the trade-off between the transmission time reduction and the accuracy of the algorithm. Define the negative logarithm of the relative error ($NLRE$) expression as

$$NLRE = -\log(|J_{gt} - J_*|/J_{gt}), \tag{4.46}$$

where J_{gt} is the true optimal value calculated directly with a convex solver, and J_* is the objective value obtained by a particular approach. Also, let us define the relative transmission time reduction (RTx) as

$$RTx = (\text{Tx}_{ADMM} - \text{Tx}_{GP})/\text{Tx}_{ADMM}, \tag{4.47}$$

where Tx_{ADMM} is the transmission time obtained when running the *Sync:Exact* algorithm, and Tx_{GP} is the transmission time obtained by any of the methods using the *STEP-GP*

algorithm. The metric RTx assumes that the *Sync:Exact* and *STEP-GP* methods use the same set of problem parameters.

We present our results in a graph where the vertical axis shows the values of RTx and the horizontal axis shows the values of $NLRE$. Each point in the graph is a tuple of transmission time reduction and accuracy, and its location shows how well it performs in terms of the trade-off between these two relative metrics. In particular, the ideal scenario is when $NLRE$ and RTx are as large as possible. Hence, we want the points to be as close to the right upper corner of the graph as possible.

4.6.5. Initial Threshold Tuning

Since the variation of the initial threshold affects the overall performance of the tested algorithms, we propose fine-tuning the initial threshold for the multiple methods proposed in this work. We consider testing 11 different initial thresholds per case, so we can capture the impact of such variation in the proposed methods. The threshold presented in Section 4.3.4 initializes its initial threshold ψ^{k_0} following the expression in (4.15). Such an initialization requires manually setting the variable ι, which indicates how proportional regarding V^{k_0} we want ψ^{k_0} to be. For all the different methods tested in this chapter, we tune ψ^{k_0} considering $\iota = [0.5, 0.6 \ldots, 1.4, 1.5]$.

4.6.6. Numerical Results Setting

In this subsection, we present the results for 10 and 30 agents when using the different query strategies proposed in this work with the threshold mechanism described in Section 4.3.4. We consider different initial threshold values following the description in Section 4.6.5. Each algorithm for the different methods was run 100 times with different

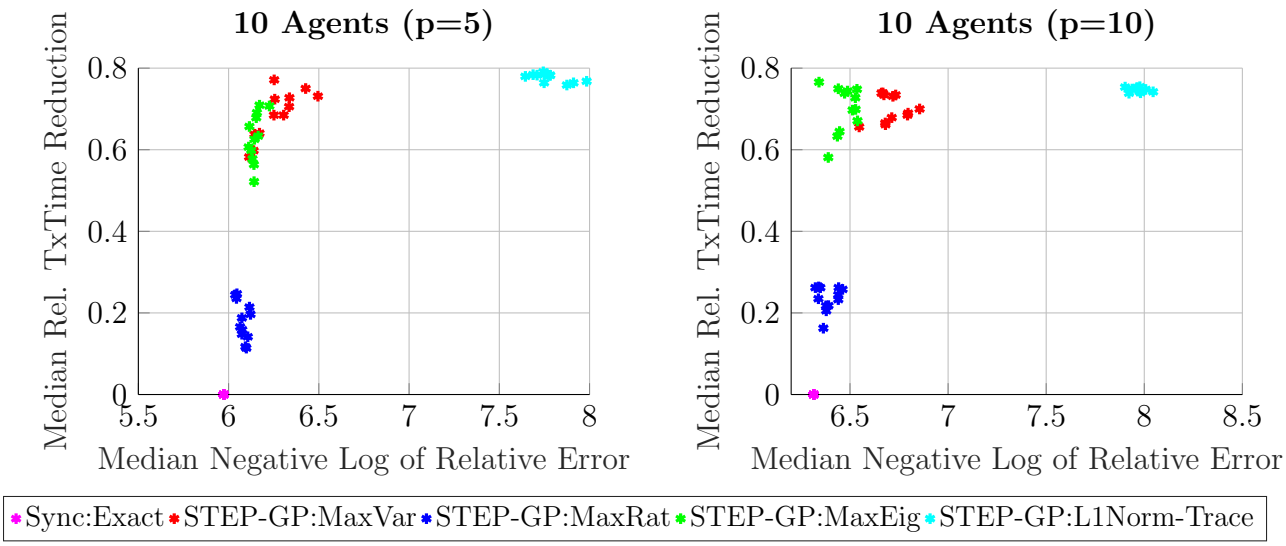

Figure 4.2. Performance trade-off between the Relative Transmission Time Reduction and the Negative Logarithm of the Relative Error for 10 Agents with variable's dimension $p = 5$ (left) and $p = 10$ (right). The plots show the 12 best-ranked tuple medians of the 100 results for different sets of parameters M_i, M_h, w_i, w_h, c_i and c_h, and for different values of α.

sets of M_i, M_h, w_i, w_h, c_i and c_h, generated as in Section 4.6.1.2. In the generated graphs, each point among the same colored cluster represents a tuple of the median values among the 100 results of the same method for the $NLRE$ and RTx metrics, as presented in Section 4.6.4.3.

The decaying threshold described in Section 4.3.4 is greatly affected by the selection of the decay rate α. For that reason, we also considered running numerical computations for different values of α on top of tuning the initial threshold. Since we consider a set of 11 initial thresholds per method, each scenario tested has 11 points per method and per value of α. The best performance of a given method might occur for a value of α that is not necessarily the same as the rest of the methods. Consequently, we present the results in Figures 4.2-4.3 as a ranking of all the median points across all different values of α tested. The ranking is done by setting a tuple as an upper bound with a value of $NLRE$

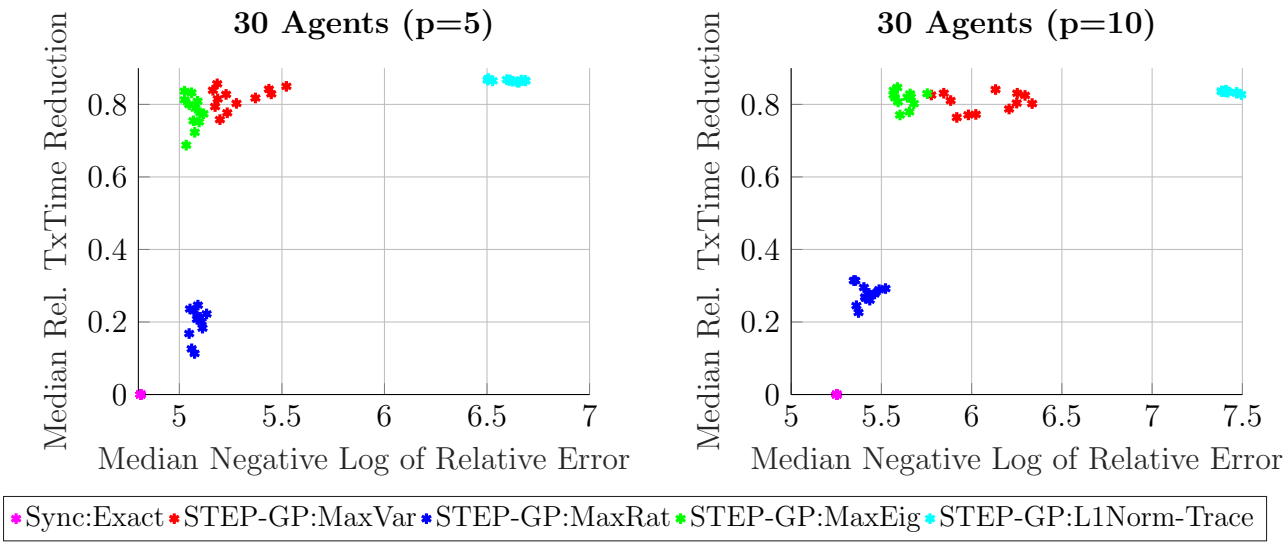

Figure 4.3. Performance trade-off between the Relative Transmission Time Reduction and the Negative Logarithm of the Relative Error for 30 Agents with variable's dimension $p = 5$ (left) and $p = 10$ (right). The plots show the 12 best-ranked tuple medians of the 100 numerical results for different sets of parameters M_i, M_h, w_i, w_h, c_i and c_h, and for different values of α.

and RTx that is higher than any of the values obtained in our results. Then we will calculate the Euclidean distance of all the median points obtained across the different values of α to the upper bound tuple. The 12 median points that attain the lowest distance are included in the graph.

This set of results considered values of $\eta = 0.2$, $\epsilon_d = 1$, $\rho = 10$, $p = 5$, an absolute tolerance value of $\epsilon^{\text{abs}} = 10^{-6}$, a relative tolerance value of $\epsilon^{\text{rel}} = 10^{-5}$, values of $\alpha = [0.95, 0.96, \ldots, 0.99]$, and $x_i^0 = \bar{y}^0 = u^0 = 0$.

4.6.7. Numerical Results for 10 and 30 Agents

Figures 4.2-4.3 (left) present the graph $NLRE$ vs. RTx for 10 and 30 agents of the median of 100 numerical results for the *Sync:Exact* and the *STEP-GP* based algorithms for the different initial thresholds considered, per each of the values considered of α when the dimension of the variables is $p = 5$, while Figures 4.2-4.3 (right) show the same infor-

mation but when the dimension of variables is $p = 10$. The presented results were selected as a consequence of a ranking of the best points in terms of the trade-off between all values tested of α. The results in all cases show three main clusters of the points presented. In the lower-left corner, the points that show the worst performance in terms of the trade-off between communication reduction and accuracy appear, which corresponds to the *STEP-GP:MaxRat* method. In the upper-left corner, with results similar to each other in all cases, appear *STEP-GP:MaxVar* and *STEP-GP:MaxEig*. These methods present a similar reduction in transmission time; however, *STEP-GP:MaxVar* presents better relative error values than *STEP-GP:MaxEig* which is showcased by the points coming from *STEP-GP:MaxVar* being closer to the ideal case. In the upper-right corner, separated from the other methods appears *STEP-GP:L1Norm-Trace* with all its points close to each other in all the graphs presented.

On the other hand, the results presented in terms of the reduction in relative transmission time in Figures 4.2-4.3 correlate with the analysis presented in Section 4.5. As the graphs show, *STEP-GP:MaxRat* presents the lowest communication reduction in all cases. The observation of the intermediate results showed that this method asked queries for each agent in around 80% of the total iterations required to reach convergence. Furthermore, the two methods based on an L2 norm confidence sphere (*STEP-GP:MaxEig* and *STEP-GP:L1Norm-Trace*) present a little more reduction in relative transmission time than the *STEP-GP:MaxVar* method. This difference is not significant if we only analyze the relative transmission time reduction metric. However, through the intermediate results, we observed that *STEP-GP:MaxEig* and *STEP-GP:L1Norm-Trace* present a lower frequency of queries, but require more iterations to converge than *STEP-GP:MaxVar*. This

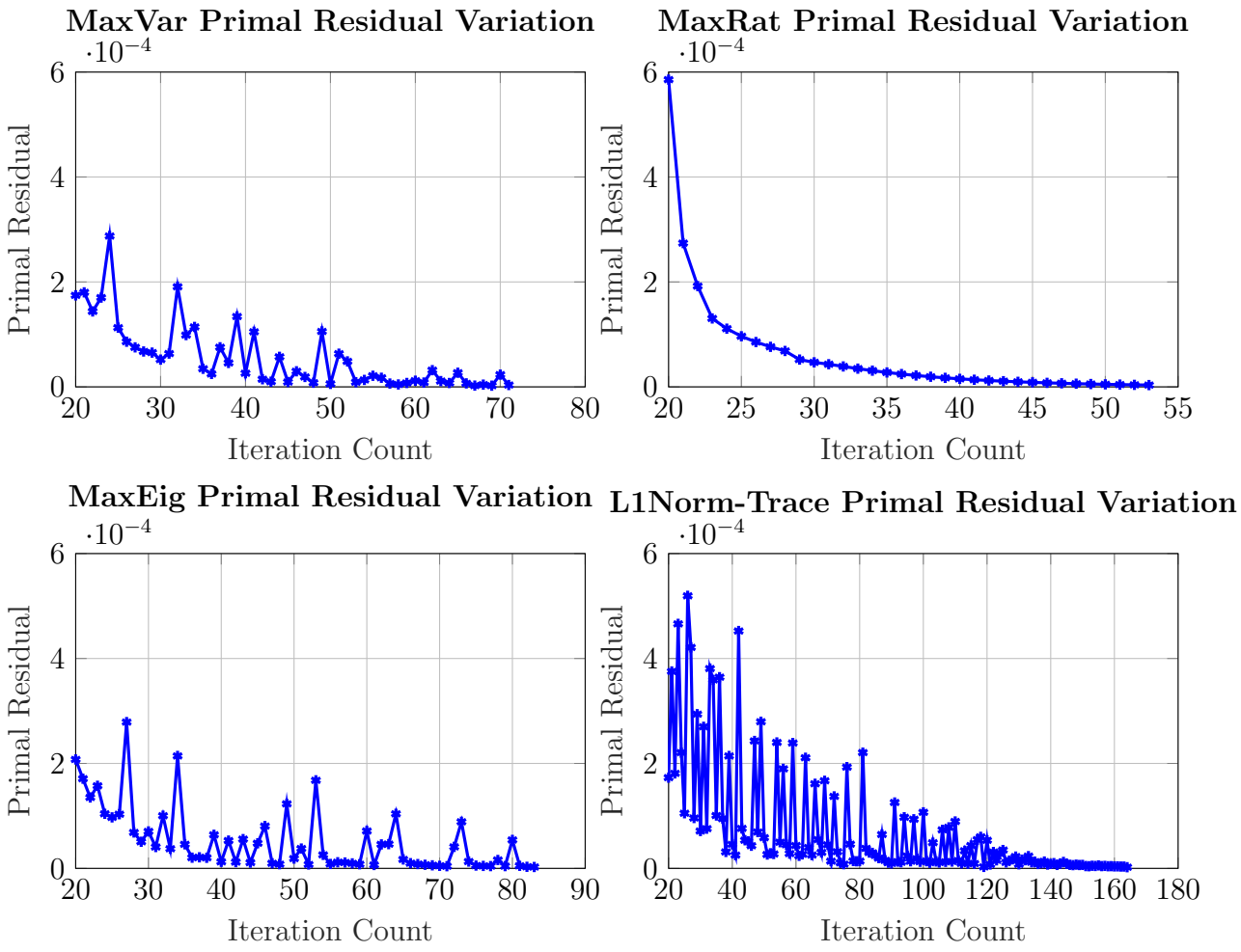

Figure 4.4. Variation of the primal residual through the iteration count for all the proposed query methods. The graphs present the test scenario for the same set of parameters M_i, M_h, w_i, w_h, c_i, and c_h of 10 agents with variables' dimension of $p = 10$, an initial threshold given by $\iota = 1$, and decay rate $\alpha = 0.97$ for all cases.

behavior is more pronounced for the *STEP-GP:L1Norm-Trace* where the frequency of queries is considerably lower but the increment in the number of iterations is also very significant. Thus, the results generated are aligned with the anticipated query behavior.

4.6.8. Empirical Convergence

In this subsection, we present results on the convergence behaviors of the proposed query methods. Figure 4.4 shows the ADMM primal residual as defined in Section 4.6.4.2 through the iteration count until convergence is reached for all methods tested. The four

graphs present the test scenario for the same set of parameters M_i, M_h, w_i, w_h, c_i, and c_h of 10 agents with the dimension of the variables $p = 10$, an initial threshold set by $\iota = 1$ and the decay rate $\alpha = 0.97$ for all cases. The figures presented show the decaying behavior of the residual until a significant drop when convergence is achieved. The main difference between methods is the speed of convergence, which is defined by the query frequency. The smaller such a frequency, the larger the convergence speed. The speed of convergence shown in Figure 4.4 for each method is aligned with the analysis presented in Sections 4.5 and 4.6.7 because we see that *STEP-GP:L1Norm-Trace* requires considerably more iterations to reach convergence than the rest of the methods, while *STEP-GP: MaxRat* requires fewer iterations than all other methods. Although only one case is presented, this trend is observed in all test scenarios considered in all our experiments presented in the previous subsections. Thus, all generated results (regardless of the parameters of the test scenario) reached convergence and each query strategy presents the same convergence speed behavior.

4.6.9. Prediction Error

In this subsection, we present statistics about how the prediction error behaves in our algorithm through all different query methods. Figure 4.5 presents two graphs showing information on the prediction error of a numerical result corresponding to agent 1 under the *STEP-GP:L1Norm-Trace* query strategy for a specific set of parameters M_i, M_h, w_i, w_h, c_i and c_h in a system of 10 agents with the dimension of the variables $p = 10$, an initial threshold set by $\iota = 1$, and decay rate $\alpha = 0.97$. To generate both graphs we calculated the real values of the Moreau Envelope and its gradient even in iterations where

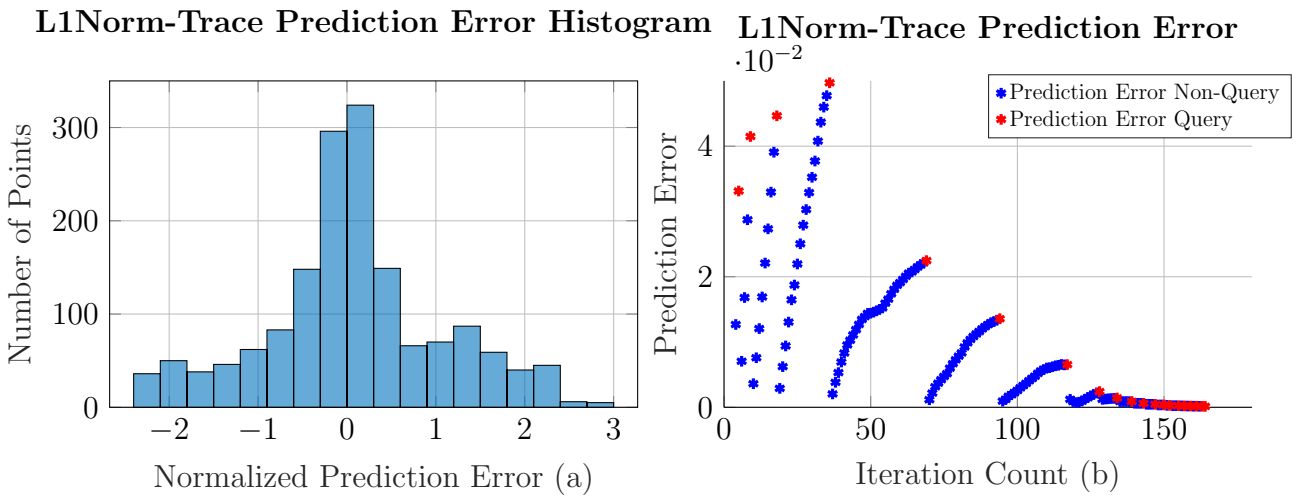

Figure 4.5. Prediction Error statistics corresponding to agent 1 under the *STEP-GP:L1Norm-Trace* query strategy for a specific set of parameters M_i, M_h, w_i, w_h, c_i, and c_h in a system of 10 agents with variables' dimension of $p = 10$, an initial threshold set by $\iota = 1$, and decay rate $\alpha = 0.97$. Graph (a) presents the histogram of the normalized prediction error, while graph (b) presents the variation of the L2 norm of the prediction error at each iteration.

a query was not requested.

In Figure 4.5 (a) we present the histogram of the normalized prediction error vector ($\epsilon^k_{i(NPE)}$), where the l^{th} entry ($l \in [1, \ldots, p+1]$) is defined as

$$\epsilon^k_{i[l](NPE)} = \left(\frac{1}{s^k_{i[l]}}\right) \left| \left[f_i^{\frac{1}{\rho}}(z^k_i); \nabla f_i^{\frac{1}{\rho}}(z^k_i) \right]_{[l]} - \mu^k_{i[l]} \right|.$$

This normalized error results in a vector generated at each iteration for each agent. To construct the presented histogram, we consider each individual component of the vector $\epsilon^k_{i(NPE)}$ as a point to be considered in the graph. Following the GP assumptions, we should expect that the discrepancy between the Moreau Envelope and its gradient with the predicted mean follows a Gaussian distribution. However, the histogram in Figure 4.5 (a) contradicts the prior expectation. This non-normality of the prediction error is also observed in other query strategies throughout different system parameters. Some cases presented histograms showing more discrepancies with respect to the expected Gaussian bell

shape than the one presented in Figure 4.5 (a). This is interesting because these results show that even though the assumed Gaussian distribution of $f_i^{\frac{1}{\rho}}(z_i^k)$ does not hold, the GP is still capable of making a good prediction with acceptable accuracy. Furthermore, this discrepancy from the initial assumption did not prevent any of the scenarios tested from reaching convergence.

On the other hand, Figure 4.5 (b) presents the variation of the L2 norm of the prediction error at each iteration for agent 1. This is defined as

$$\epsilon_{i[PE]}^k = \left|\left|\left[f_i^{\frac{1}{\rho}}(z_i^k); \nabla f_i^{\frac{1}{\rho}}(z_i^k)\right] - \mu_i^k\right|\right|_2 .$$

This metric generates a single point per iteration, so the presented graph shows the variation of the prediction error over the algorithmic iterations. Figure 4.5 (b) also makes a differentiation between iterations in which a query was made (green points) and iterations in which no query was made (blue points). The decaying behavior of the prediction error is clearly seen in the graph with a significant drop closer to convergence. This behavior is desirable because we want our prediction to become more accurate through the algorithmic iterations, which is a favorable condition to be confident not only that we reach convergence but that we converge to a good solution. Furthermore, the figure shows a bursting behavior between intervals, where we see an increment in the prediction error during the interval where no query was made and an abrupt drop once a query is requested. This prediction error behavior is observed for all agents through all the different test scenarios and different query strategies.

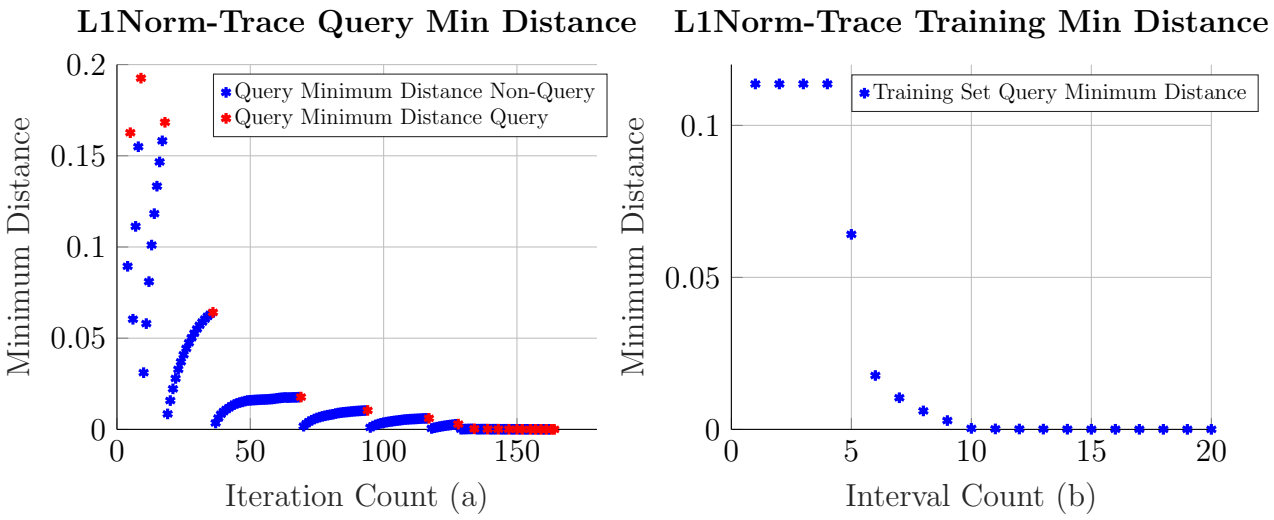

Figure 4.6. Distances between generated query points for a specific set of parameters M_i, M_h, w_i, w_h, c_i, and c_h in a system of 10 agents with variables' dimension of $p = 10$, an initial threshold set by $\iota = 1$, and decay rate $\alpha = 0.97$. Graph (a) presents the measurement of the minimum distance between a new query vector and all query vectors already in the training set. Graph (b) presents the minimum query distances between query points that are already part of the training set only.

4.6.10. Query Dynamics

In this subsection, we present information on the distances between the queries z_i^k generated at each iteration compared to the previous query points included in the GP training set. Figure 4.6 (a) presents the measurement of the minimum distance between a new query vector and all query vectors already in the training set. This distance is defined as

$$d(z_i^k, Z^k) = \min\{d(z_i^k, z) : z \in Z_i^k\},$$

where Z_i^k is the set containing the queries within the GP training set for agent i until iteration k and $d(\cdot)$ is the distance function. Since each generated z_i^k is a vector, the distance function considered is $d(z_i^k, Z^k) = \|z_i^k - z\|_2$ where $z \in Z_i^k$. Figure 4.6 (a) presents a differentiation between iterations in which a query was made (green points) and iterations

in which there was no query (blue points). The results show that the distance between the queries throughout the iterations tends to become smaller as the iteration process approaches convergence. This is correlated with the patterns observed in Figure 4.5 (b), where the prediction error is smaller when the algorithm is closer to convergence. The closer the query points are to the end of the algorithm run, the more points are trained in GP around a close vicinity, thus considerably reducing the uncertainty of the prediction. Furthermore, the behavior of the minimum distance between queries presented in Figure 4.6 (a) presents a similar bursting behavior to that observed for the prediction error in Figure 4.5 (b).

On the other hand, Figure 4.6 (b) presents the minimum query distances between the query points already included in the training set. Only when a new point is added to the training set is this minimum distance recalculated. This distance is defined as

$$d(z, x) = \inf\{d(z, x) : z, x \in Z_i^k, z \neq x\},$$

where $d(.)$ once again is defined as $d(z, x) = \|z - x\|_2$. The graph in Figure 4.6 (b) presents a new point when a query is made, so each point presented represents an interval after a period of iterations where no query was made. Similarly to the results presented in Figure 4.6 (b), the distance between the query points also decreases closer to convergence. However, in the case where we only compare points that are part of the training set, we do not see increasing variations at any point.

4.6.11. Overall Remarks

The presented results across different initial parameters showed that the joint query method *STEP-GP:L1Norm-Trace* is the method that achieved better trade-off perfor-

mance among all query strategies tested. An observation we made during the numerical computations is that such a method tends to reduce the required queries considerably; however, it does not require extensive communication rounds to obtain good values for the $NLRE$ metric. Compared to the other methods tested, for similar values of total transmission time, the *STEP-GP: L1Norm-Trace* method usually produces a global ADMM solution closer to the true solution. In contrast, the *STEP-GP:MaxRat* method proved to be the one with the worst trade-off performance among all tested methods. Although the other individual query strategies showed similar behavior, it was *STEP-GP:MaxVar* that showed a better overall trade-off performance compared to *STEP-GP:MaxEig*. In addition, the results obtained were consistent across all the different numerical computation cases presented. The querying behavior observed during numerical computations correlates with the previous analysis, resulting in an anticipated querying behavior of the proposed methods.

The results presented showed that the more complex querying strategy can achieve the best performance. This outcome agrees with the intuitive idea that the method closer to the general querying framework should achieve better performance. On the other hand, the individual query methods, despite their simplicity, were able to maintain an acceptable accuracy while reducing the transmission time considerably. Thus, the individual strategies *STEP-GP:MaxVar* and *STEP-GP:MaxEig* are viable options in scenarios where the computation cost needs to be as low as possible.

4.7. Conclusion to Chapter 4

Distributed optimization methods, such as ADMM, generally incur an excessive undesired communication overhead. In this context, the use of Gaussian processes has proven to be effective in learning the unknown proximal operators of the agents. Therefore, the coordinator can predict the solutions to the local proximal minimization sub-problems, requiring fewer queries to the agents, which leads to a significant reduction in communication. However, the extent of the achievable reduction in communication depends in part on the mechanism through which the coordinator decides if communication with the agents is needed. For that reason, this work proposed several query strategies to decide whether the coordinator should send queries to the agents in a particular iteration when running the *STEP-GP* algorithm based on the notion of a general querying framework. Such an ideal mechanism solves a constrained optimization problem by balancing two opposing criteria, which are to maximize the communication reduction while minimizing the error of the final solution obtained. Motivated by this constrained optimization problem and an alternative expression of the regular ADMM updates that showcases the inherent coupling between agents, we proposed a joint query strategy consisting in minimizing a convex communication cost restricted by the trace of the joint uncertainty of the ADMM variables. On the other hand, to reduce the computational burden added to our algorithm, we proposed different individual query strategies for each agent using an individual uncertainty measure to determine whether the prediction is reliable enough to skip a communication round. The numerical results of solving a sharing problem with quadratic cost functions showed the different performances of the proposed methods in terms of the

trade-off between the reduction of communication cost and the loss of accuracy in solving the optimization problem. In particular, the proposed joint query method achieved better trade-off performance than the independent query strategies. Our next research steps include testing our proposed framework in more complex applications, where we have more challenging objective functions, and convergence analysis of all query methods.

Chapter 5. LGP with Adaptive Quantization Resolution

In Chapter 2 we solved a distributed centralized optimization problem through ADMM with a learning component to skip communication rounds, where communications between the coordinator and agents are quantized. Because we proposed to use a uniform quantizer that adapts its mid-value and window length with the statistics given by the predictor, we were able to characterize the quantization error statistics. These statistics do not follow a Gaussian distribution, making the GP assumptions invalid. Therefore, we proposed an alternative regression method (based on GP) following the concept of Linear Minimum Mean Square Error (LMMSE) estimation. Our proposed method resulted in the integration of ADMM with our new proposed regression method affected by uniform quantization, where the impact of the quantization error was addressed and mitigated. The study presented in Chapter 4 dealt with the same framework presented in [23] where quantization was not considered. In this scenario, GP was used as the learning method and our work focused on the mechanism used to determine whether the coordinator should communicate with the agents or not. This study resulted in the proposition of an ideal query method that served as a framework to derive different individual and joint query strategies.

This chapter aims to integrate these two previous studies by adding a new component: A quantization scheme that allows the bit resolution to be varied at each iteration. This new component is motivated by the fact that not all agents have the same requirements or deal with the same amount of uncertainty in each round. The quantization scheme used in Chapter 2 assigned the same resolution to all agents and did not vary the given value during the execution of our algorithm. We propose generating a mecha-

nism that allows the coordinator to increase or decrease the quantization resolution of each agent according to their behavior. Furthermore, the query decision and quantization resolution allocation mechanisms are aligned with the conditions considered in Chapter 3, allowing us to apply the derived convergence analysis directly to the approach proposed in this chapter.

Chapter Organization: We begin with the formulation of the problem in Section 5.1. The general query and quantization allocation framework is presented in Section 5.2. We present our proposed joint query and resolution allocation mechanism in Section 5.3, followed by our proposed individual strategy in Section 5.4. The numerical results are presented in Section 5.5, and the conclusions are presented in Section 5.6.

5.1. Problem Formulation

This chapter addresses a collaborative optimization scenario that involves n agents and a central coordinator, similar to the setup explored in Chapter 2. In this setup, the objective is to minimize a global cost function comprising individual strongly convex cost functions $f_i : \mathbb{R}^p \to \mathbb{R}$ where each agent has local decision variables $x_i \in \mathbb{R}^p$, alongside a convex shared cost function $h : \mathbb{R}^p \to \mathbb{R}$, formulated as follows:

$$\text{minimize} \quad \sum_{i=1}^{n} f_i\left(x_i\right) + h\left(\sum_{i=1}^{n} x_i\right). \tag{5.1}$$

Here, f_i is only accessible to the corresponding agent. Furthermore, communication for solving (5.1) is restricted to exchanges between the coordinator and the agents, with no direct interactions between the agents themselves.

The optimization problem (5.1) is addressed using the Alternative Direction

Method of Multipliers (ADMM) with proximal operators following the query response mechanism presented in Section 2.1.

5.1.1. Quantization Statistics with LGP Overview

In Chapter 2 we further reduced communication overhead compared to [23] by considering uniform quantization in communications between agents and coordinator. In that chapter, we consider a quantization scheme that adapts the quantizer by setting the mid-value to the conditional mean given by GP, and the window length to be proportional to the diagonal of the covariance matrix given by LGP.

When dealing with the total uncertainty introduced, the source of the uncertainty is either the prediction error or the quantization error. The LGP gives its uncertainty measurement per agent through the conditional covariance matrix $\Sigma_i^k(z_i^k)$. The diagonal of such a matrix gives the variance of the prediction uncertainty. Following the derivations made in Chapter 2 the uniform quantization error variance is:

$$\mathrm{Var}(\epsilon_{\mathbb{Q}}) = \frac{1}{12}q^2, \tag{5.2}$$

where q is the quantizer's window length defined as $q = \frac{2c}{2^{b_i^k}}\sigma_i^k$ (q is a vector). This leads to the expression,

$$\mathrm{Var}(\epsilon_{\mathbb{Q}}) = \frac{c^2}{3(2^{2b_i^k})}s_i^k, \tag{5.3}$$

for some given $c > 0$ and $s_i^k = \mathrm{diag}(\Sigma_i^k(z_i^k))$.

5.1.2. Query-Response Dynamics

In Figure 5.1, we present one round of the proposed algorithm for a network of 2 agents. This figure uses the notation defined throughout Chapter 2. The LGP regression block named proxLGP refers to the prediction of $f_i^{\frac{1}{\rho}}(z_i^k)$ and $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ as presented in

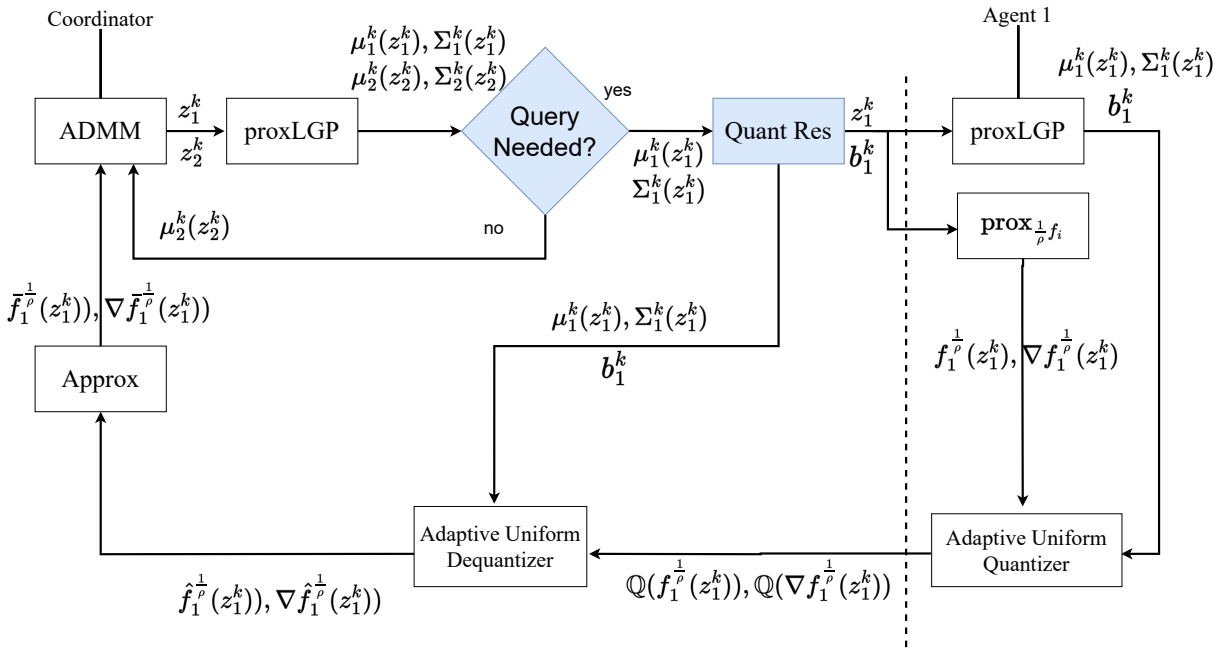

Figure 5.1. Flow diagram of a query and response between the coordinator and an agent in the proposed approach. The enhancements contributed by this chapter, compared with the approach in Chapter 2, are highlighted in the blue-shaded boxes.

Theorem 1 in Chapter 2. The coordinator has a corresponding proxLGP for each agent, which is trained on its past query data with the agent. The coordinator first calculates the query variables z_i^k for each agent and uses them as input to the agent's proxLGP. Using the covariance matrices $\Sigma_i^k(z_i^k)$ given by the proxLGPs, the coordinator decides which agents to query. Then, for the agents set to be queried, the coordinator assigns a quantization resolution to each one. In Figure 5.1, agent 1 is set to be queried, so the coordinator sends z_1^k and b_1^k to the agent, which solves its proximal minimization problem, represented by block $\mathbf{prox}_{\frac{1}{\rho}f_i}$. Subsequently, agent 1 quantizes $f_1^{\frac{1}{\rho}}(z_1^k)$ and $\nabla f_1^{\frac{1}{\rho}}(z_1^k)$, $\nabla f_2^{\frac{1}{\rho}}(z_2^k)$ adapting its mid-value, window length, and bit resolution. The quantized response $\left\{\left(\mathbb{Q}\big(f_i^{1/\rho}(z_i^k)\big), \mathbb{Q}\big(\nabla f_i^{1/\rho}(z_i^k)\big)\right)\right\}$ of agent 1 is sent back to the coordinator, that uses a similar dequantization process based on the same predictive mean $\mu_1^k(z_1^k)$,

covariance matrix $\Sigma_1^k(z_1^k)$, and quantization resolution b_i^k to obtain the dequantized response $\{\hat{f}_i^{1/\rho}(z_i^k), \nabla \hat{f}_i^{1/\rho}(z_i^k)\}$. Finally the coordinator uses the approximated values $\{\bar{f}_i^{1/\rho}(z_i^k), \nabla \bar{f}_i^{1/\rho}(z_i^k)\}$, coming from the estimation process according to Theorem 2 in Chapter 2, for the ADMM updates. Meanwhile, for agent 2, which is not queried, the coordinator uses the corresponding predicted values $\mu_2^k(z_2^k)$ from its proxLGP to perform the ADMM updates.

5.1.3. ADMM Updates with LGP and Adaptive Quantization

Following the query-response mechanism shown in Figure 5.1, the ADMM formulations are adjusted to integrate the LGP regression. Initially, we define the communication decision variable for agent i in iteration k as

$$\gamma_i^k = \begin{cases} 1, & \text{if agent } i \text{ is queried} \\ 0, & \text{otherwise.} \end{cases} \tag{5.4}$$

When $\gamma_i^k = 1$, the query z_i^k is sent to agent i to acquire the quantized value of $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ given by $\nabla \hat{f}_i^{1/\rho}(z_i^k)$. On the contrary, when $\gamma_i^k = 0$, we utilize the predicted value $\mu_i^k(z_i^k)$ from the LGP regression. Consequently, we introduce the received value β_i^k as

$$\beta_i^k = \gamma_i^k \nabla \hat{f}_i^{\frac{1}{\rho}}(z_i^k) + (1 - \gamma_i^k)\mu_i^k(z_i^k). \tag{5.5}$$

Subsequently, the ADMM formulations for the sharing problem in (5.1) as:

$$\begin{aligned} x_i^{k+1} &= z_i^k - (1/\rho)\beta_i^k \\ \bar{y}^{k+1} &= \underset{\bar{y} \in \mathbb{R}^p}{\arg\min} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^{k+1} - u^k\|^2 \right\} \\ u^{k+1} &= u^k + \bar{x}^{k+1} - \bar{y}^{k+1}. \end{aligned} \tag{5.6}$$

This chapter concentrates on the effective execution of query decision-making when dealing with adaptive uniform quantization resolution allocation. The following section presents the general framework used to derive our proposed solutions for making the communication decision and assigning the quantization resolution optimally.

5.2. General Framework

The primary aim of integrating regression and quantization into the ADMM algorithm to solve distributed optimization problems is to mitigate communication overhead. However, it is crucial to ensure that this incorporation does not significantly impede algorithm convergence or compromise the accuracy of the optimization solution. The pivotal elements in ADMM updates when LGP is used, as illustrated in (5.6), are the variable β_i^k and the quantization resolution b_i^k. The variable β_i^k assumes the role of the quantized gradient $\nabla \hat{f}_i^{\frac{1}{\rho}}(z_i^k)$ of the Moreau envelope or its predicted counterpart, dependent on γ_i^k. In the case where communication is required, a quantization resolution is assigned to each communicating agent. In (5.6), the ensemble of x^{k+1}, $\bar{y}^k$, and u^{k+1} can be construed as a high-dimensional vector trajectory toward the global solution. This trajectory is influenced by β_i^k and b_i^k, which are contingent on the communication decision variable γ_i^k, as defined in (5.4) and (5.5), thus affecting both the accuracy of the LGP regression and the optimization performance. Consequently, the mechanism to determine γ_i^k and b_i^k will inherently impact the overall performance of communication and optimization. In the absence of a robust and systematic mechanism for the coordinator to discern when to dispatch queries to agents and allocate the quantization bits for the communicating ones, the ADMM algorithm may either necessitate excessive iterations to converge or fail to achieve

convergence altogether.

We propose a systematic approach in which the query decision and the bit resolution allocation are jointly analyzed and performed as one complex task. This means that the trade-off between uncertainty and communication expenditure does not depend only on the binary communication decision of each agent γ_i^k but also depends on the quantization resolution b_i^k assigned to each agent. Intuitively, we can model a minimization of the form

$$\begin{aligned} \text{minimize} \quad & \text{Comm}(\gamma^k, b^k), \\ \text{subject to} \quad & b^k \in \mathcal{N}^n, \\ & \gamma^k \in \{0, 1\}, \\ & \text{Uncert}(\gamma^k, b^k) < \psi^k, \end{aligned} \tag{5.7}$$

where γ^k is a vector containing the binary communication variable of each agent, b^k is a vector containing the quantization resolution of each agent at the present iteration, $\text{Uncert}(\gamma^k, b^k)$ is the uncertainty function, $\text{Comm}(\gamma^k, b^k)$ is a communication function, and ψ^k is a given threshold varying at each iteration.

This general framework relies on two opposing criteria: 1) reduce communication overhead and 2) maintain the convergence and accuracy of the ADMM algorithm. This is conceptually similar to the general querying framework presented in Section 4.2 but has larger implications. The idea proposed in (5.7) has critical differences from the general framework in Section 4.2, such as:

- The contribution of each agent to the uncertainty function without quantization either came from the prediction error or was zero because the real value was used. In (5.7) each agent will always contribute to the uncertainty because if we communicate, the quantization error will contribute to the overall uncertainty.

- The communication cost in (5.7) is now a more complex cost because the payload of the information shared now affects the communication overhead. This was not a variable in Section 4.2 because the payload of the shared information was the same for all agents. In this new setting, agents respond with different payload sizes.

Typically, solving the optimization problem (5.7) requires a combinatorial approach due to the n binary variables $\gamma_{i\ i=1,\cdots,n}^{k}$. However, computational costs can become prohibitive as the number of agents increases. Consequently, in this chapter, our goal is to explore methods for addressing (5.7) while adhering to specific communication and uncertainty functions, without relying on combinatorial techniques.

5.3. Proposed Joint Approach

In this section, we introduce a collective approach to make the communication decision and assign the quantization resolution based on the general framework presented in the previous section. In this approach, the uncertainty measure defined in Equation (5.7) is the sum of the diagonal elements of the joint covariance matrix of the ADMM variables influenced by the LGP regression. In the following subsection, we provide the measure of uncertainty used in our algorithm.

5.3.1. Uncertainty Expression when using the LGP approach

The analysis presented in Section 4.3.1 justifies using the trace of the covariance matrix given by the LGP regression as the uncertainty function. In case we do not consider quantization, this function is the trace of the LGP conditional covariance matrix $\Sigma_i^k(z_i^k)$. However, due to the inclusion of quantization, we have to account for the uncertainty introduced by the quantization process. Following the study made in Section 2.3.1, the gradient of the Moreau Envelope $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ is quantized element by element, where each element of $\nabla f_i^{\frac{1}{\rho}}(z_i^k)$ is quantized by a uniform quantizer having its mid-value set by

$\mu_i^k(z_i^k)$ and its corresponding element of the window length vector defined as

$$q_i^k = \frac{2c}{2^{b_i^k}} \sqrt{\text{diag}(\Sigma_i^k(z_i^k))}, \tag{5.8}$$

where $c > 0$ is a given constant. Therefore, following Proposition 1 in Section 2.3.1 and the adaptation of the window length in (5.8), we can approximate the quantization error of a given agent in iteration k given by $\epsilon_{i\mathbb{Q}}^k$ with its l-th entry $\epsilon_{i\mathbb{Q}[l]}^k$, $l \in [1, \ldots, p]$, to follow a uniform distribution given by $\epsilon_{i\mathbb{Q}[l]}^k \sim \mathcal{U}[-q_{i[l]}^k/2, q_{i[l]}^k/2]$. Consequently, the variance vector of the uniform quantization error for a given agent at iteration k is as follows:

$$\text{Var}(\epsilon_{i\mathbb{Q}}^k) = \frac{1}{12}(q_i^k)^2. \tag{5.9}$$

Thus, the overall quantization uncertainty is given by the summation of each element of the vector $\text{Var}(\epsilon_{i\mathbb{Q}}^k)$. Then, we can formulate an overall uncertainty function for all agents depending on the decision variable γ_i^k and each agent's quantization resolution b_i^k as:

$$\text{Uncert}(\gamma^k, b^k) = \sum_{i=1}^{n} \gamma_i^k \frac{\theta}{2^{2b_i^k}} \text{tr}\left(\Sigma_i^k(z_i^k)\right) + (1 - \gamma_i^k) \text{tr}\left(\Sigma_i^k(z_i^k)\right), \tag{5.10}$$

where $\theta = \frac{c^2}{3}$ and $\text{tr}(\cdot)$ is the trace operator.

5.3.2. Proposed Joint Query and Resolution Allocation Method

Using the general framework outlined in Section 5.2, we suggest employing the total number of bits transmitted in the current iteration as the metric for the cost of communication. This metric is denoted as the multiplication of the number of agents queried in the current iteration and each of their assigned quantization resolutions. On the other hand, the uncertainty function $\text{Uncert}(\gamma^k, b^k)$ is constructed using the trace of the conditional covariance matrix presented in (5.10). Therefore, the specific form of the general optimization problem (5.7) is:

Algorithm 3 Joint Query and Quantization Resolution Allocation

Require: Each agent's value of $\Sigma_i^k(z_i^k)$, the initial value and adaptation of ψ^k, set all $\gamma_i^k \leftarrow 1$.

1: Sort each $\text{tr}(\Sigma_i^k(z_i^k))$ from smallest to greatest, and store the sorting as $m^k = [m_1^k, m_2^k \dots m_n^k]$.
2: Define idx_i^k, as a flag corresponding to the position in the sorting that corresponds to the value of m_i^k.
3: Calculate the cumulative sum vector of m^k as $\text{CM}^k = cumsum(m^k)$.
4: Define n_s as the largest value of CM^k such that $\text{CM}_{n_s}^k \leq \psi^k$, and set $\gamma_j^k \leftarrow 0$, where $j = [1, \dots, n_s]$.
5: **if** $n_s = n$ **then**
6: Terminate
7: **else**
8: Run Algorithm 4
9: **end if**

$$\begin{aligned} \text{minimize} \quad & \|\gamma^k\|_1 \sum_{i=1}^{n} (1-\gamma_i^k) b_i^k \\ \text{subject to} \quad & \gamma_i^k \in \{0,1\}, 1 \leq i \leq n, \\ & b_i^k \in \mathcal{N}, \\ & \sum_{i=1}^{n} \gamma_i^k \frac{\theta}{2^{2b_i^k}} \text{tr}\left(\Sigma_i^k(z_i^k)\right) + (1-\gamma_i^k) \text{tr}\left(\Sigma_i^k(z_i^k)\right) \leq \psi^k, \end{aligned} \tag{5.11}$$

where the threshold ψ^k varies with each iteration. The rationale behind Equation (5.11) is to select the smallest set of agents to query with the least possible quantization resolution while ensuring that the joint uncertainty remains below the threshold ψ^k. This guarantees a high probability that the uncertainty remains within a desired bound.

Subsequently, we introduce an efficient solution to the problem stated in (5.11) without relying on combinatorial methods, leveraging the convexity and linearity inherent in the communication and uncertainty functions considered. This approach is constructed by solving the problem in (5.11) in two parts. The first part focuses only on determining the value of each of the communication decision variables γ_i^k, while the second assigns the

quantization resolution b_i^k to each communicating agent. The approach revolves around the following steps:

1. Define a global threshold ψ^k and a quantization threshold $\psi^{Q[k]}$ both decaying at each iteration. Also, define a minimum and maximum allowed quantization resolution.

2. Initiate the search for a query set with the scenario where communication cost is at its peak while uncertainty is at its minimum.

3. Then, we calculate the contribution to the sum of all traces of each agent, where the ones that contribute the least to the overall sum are the first candidates not to be queried in the current round.

4. Rather than examining every potential combination, we analyze the threshold ψ^k against the sum of the traces each time the next candidate is poised to skip communication until the constraint is satisfied with the largest possible number of agents skipping communication. This number is defined as n_s.

5. We calculate the remainder of the total threshold by subtracting the sum of the uncertainty of the noncommunicating agents to ψ^k. This remainder is defined as ϵ_s.

6. We check feasibility by comparing the minimum possible uncertainty (when all communicating agents use the maximum possible resolution) with ϵ_s. If feasibility is not reached, we set another agent to communicate (increasing the value of n_s by 1) according to its contribution to the total uncertainty and recalculate ϵ_s until the problem is feasible.

7. Assign the quantization resolution of the communicating agents as $b_i^k = \left\lceil 0.5 \log_2 \left(\left(\frac{\theta(n-n_s)}{\psi^{Q[k]}} \right) \operatorname{tr} \left(\Sigma_i^k(z_i^k) \right) \right) \right\rceil$, where n_s is the number of communicating agents. If b_i^k is beyond the boundaries set by the minimum and maximum resolutions, then we set it to the value of its closest boundary.

The details of the proposed approach are presented in Algorithm 3 and Algorithm 4.

5.3.3. Threshold ψ^k and $\psi^{Q[k]}$ Mechanism

During the execution of the ADMM algorithm, the uncertainty associated with the LGP regression tends to decrease as the algorithm approaches convergence. This reduc-

Algorithm 4 Joint Quantization Resolution Allocation

Require: The values of the minimum resolution l and maximum resolution h, the value of n_s, the vectors m^k and γ^k, and the initial value and adaptation of $\psi^{Q[k]}$

Calculate the slack: $\epsilon_s \leftarrow \psi^k - \sum_{i=1}^{n_s} m_i^k$.
for $j = n_s, \ldots, 1$ **do** ▷ Go through a For Loop of the possible values we can decrease n_s
  if $\sum_{i=n_s+1}^{n} \frac{1}{2^{2h}} m_i^k \leq \epsilon_s/\theta$ **then** ▷ Check feasibility
    break
  else ▷ if not feasible decrease n_s and check feasibility again
    $n_s \leftarrow n_s - 1$, set $\gamma^k_{\mathrm{idx}_j^k} \leftarrow 1$
    $\epsilon_s \leftarrow \psi^k - \sum_{i=1}^{n_s} m_i^k$
  end if
end for
Define $r = [n_s + 1, n_s + 2, \ldots, n]$.
for $j = r$ **do**
  $b^k_{\mathrm{idx}_j^k} \leftarrow \left\lceil 0.5 \log_2 \left(\left(\frac{\theta(n-n_s)}{\psi^{Q[k]}} \right) m_j^k \right) \right\rceil$
  if $b^k_{\mathrm{idx}_j^k} < l$ or $b^k_{\mathrm{idx}_j^k} > h$ **then** ▷ Saturate if the $b^k_{\mathrm{idx}_j^k}$ is beyond boundaries
    Set $b^k_{\mathrm{idx}_j^k} \leftarrow l$ or $b^k_{\mathrm{idx}_j^k} \leftarrow h$
  end if
end for

tion in uncertainty is attributable to the availability of more training data obtained from responses to queries, thereby enhancing the accuracy of predictions. Consequently, it is advisable for the threshold considered to decrease over successive ADMM iterations. To address this, we propose a mechanism for decreasing the threshold that is based on both the iteration count and k_0, which denotes the iteration when the LGP regression is used for the first time.

Initially, we define the threshold at iteration k_0 as:

$$\psi^{k_0} = \iota V^{k_0}, \tag{5.12}$$

where V^{k_0} represents the total uncertainty variable utilized by the query method (in this instance, $\sum_{i=1}^{n} \mathrm{tr}\left(\Sigma_i^{k_0}(z_i^{k_0})\right)$), and ι, predetermined, is a scalar ranging between 0 and 1. Subsequently, given a preselected decay rate $\alpha \in (0, 1)$, at a later iteration $k > k_0$, the

threshold is updated as follows:

$$\psi^k = \psi^{k_0}\alpha^{k-k_0}. \tag{5.13}$$

The quantization threshold $\psi^{Q[k]}$ is defined and adapted following the same mechanism. In this case, $\psi^{Q[k]} = \psi^{Q[k_0]}\alpha_Q^{k-k_0}$, where $\psi^{Q[k_0]} = \iota_Q V^{Q[k_0]}$, $V^{Q[k_0]} = \sum_{i=1}^{n} \frac{\theta}{2^{2l}} \operatorname{tr}\left(\Sigma_i^{k_0}(z_i^{k_0})\right)$, l is the minimum quantization resolution, and ι_Q and α_Q are scalars that vary between 0 and 1.

5.3.4. Convergence Analysis

The joint method proposed in this section uses the same uncertainty measure and meets the conditions presented in the convergence analysis in Chapter 3. For that reason, if there is no bound on the number of bits that can be assigned for quantization, then the convergence analysis in Section 3.5, which concludes with Theorem 8, applies to the method presented in this section.

In case the number of bits that can be assigned for quantization is bounded, then our proposed approach convergence analysis is analogous to the one presented in Section 3.6. This analysis concludes that the expectation of the ADMM residual is always bounded and that this bound decreases with each iteration.

5.4. Proposed Individual Approach

In this section, we simplify the problem posed in (5.11). The idea of this approach is that each agent makes its communication decision and assigns the quantization resolution without taking into account the decisions of the other agents. If we consider doing our communication decision and quantization bits assignment individually, the two sources of uncertainty are mutually exclusive. This approach significantly decreases the computa-

tional complexity compared to the overarching method outlined in Section 5.2. However, it overlooks the influence of an agent's decision on the overall prediction error introduced into the system. Nonetheless, by constraining the uncertainty of each agent per iteration, we guarantee that the prediction error minimally impacts the performance of the ADMM algorithm. While this strategy may not match the rigor of the joint method, its simplicity makes it appropriate for scenarios prioritizing minimal computational expense.

In this individual method, each agent makes its own querying and resolution allocation decisions independently reflected in the agent's corresponding binary decision variable γ_i^k and resolution bits b_i^k respectively. The concept behind this approach is that for each agent i, the coordinator decides whether to refrain from sending a query to that agent if the probability of an estimation error, both for the Moreau Envelope and its gradients, falls within an acceptable individual threshold ψ^k. In the event that communication is required, the quantization resolution is determined by the variance of the quantization error compared with a decaying threshold $\psi_i^{Q[k]}$. By adopting this approach, we circumvent the minimization problem outlined in (5.11), determining each γ_i^k and b_i^k by comparing the estimated errors of each agent to individual thresholds. This method considers the following steps to be performed for each agent:

1. Define the local threshold ψ_i^k and a local quantization threshold $\psi_i^{Q[k]}$ both following the mechanism presented in Section 5.3.3. Also, define a minimum and maximum allowed quantization resolution.

2. In case the trace of $\Sigma_i^k(z_i^k)$ is below ψ_i^k, then agent i is set to not communicate so the conditional mean $\mu_i^k(z_i^k)$ is used in the ADMM algorithm updates.

3. In case the trace of $\Sigma_i^k(z_i^k)$ is greater than ψ_i^k, then agent i is set to communicate and its resolution is calculated by comparing the variance of the quantization error to the quantization threshold $\psi_i^{Q[k]}$. If b_i^k is beyond the boundaries set by the minimum and maximum resolutions, then we set it to the value of its closest boundary.

Algorithm 5 Individual Query and Quantization Resolution Allocation

Require: The value of $\Sigma_i^k(z_i^k)$, the initial value and adaptation of ψ_i^k and $\psi_i^{Q[k]}$, and the values of the minimum resolution l and maximum resolution h.

if $\operatorname{tr}\left(\Sigma_i^k(z_i^k)\right) < \psi_i^k$ **then**
    $\gamma_i^k = 0$
else
    $\gamma_i^k = 1$
    $b_i^k = \left\lceil 0.5 \log_2 \left(\left(\frac{\theta}{\psi_i^{Q[k]}} \right) \operatorname{tr}(\Sigma_i^k(z_i^k)) \right) \right\rceil$
    if $b_i^k < l$ or $b_i^k > h$ **then**
        Set $b_i^k \leftarrow l$ or $b_i^k \leftarrow h$
    end if
end if

More details of the individual method are presented in Algorithm 5. The following section presents numerical results to validate and compare our proposed methods.

5.5. Numerical Experiments

In this section, we assess the methods proposed in this study by addressing a sharing problem in which the agent's sub-problems are quadratic. We proceed by detailing the specifics of the considered sharing problem, outlining the numerical experiment settings, and presenting the results obtained.

5.5.1. Sharing Problem

5.5.1.1. Problem Definition

Our evaluation is based on a sharing problem inspired by the application presented in [10]. In this scenario, we address a dynamic sharing problem where the problem's variables remain fixed and do not change at each algorithmic step, unlike the original formulation. We define the following sharing problem:

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n}(x_i-\theta_i)^T\Upsilon_i(x_i-\theta_i)+\zeta\|\sum_{i=1}^{n}y_i\|_1 \\ \text{subject to} \quad & x_i-y_i=0 \end{aligned} \tag{5.14}$$

Here, $x_i, y_i \in \mathbb{R}^p$, $\theta_i \in \mathbb{R}^p$, $\Upsilon_i \in \mathbb{R}^{p\times p}$ are positive definite matrices, and $\zeta > 0$ are given problem parameters. The generation of parameters θ_i and Υ_i is as presented in Section 2.5.1.2 in Chapter 2.

5.5.1.2. Solution of the Sharing Problem with ADMM

The problem described in (5.14) resembles (5.1) from Section 5.1, and the corresponding ADMM updates are summarized as follows:

$$\begin{aligned} x_i^{k+1} &= \underset{x_i\in\mathbb{R}^p}{\arg\min}\left\{f_i(x_i)+(\rho/2)\|x_i-z_i^k\|_2^2\right\} \\ \bar{y}^{k+1} &= \underset{\bar{y}\in\mathbb{R}^p}{\arg\min}\left\{\zeta\|n\bar{y}\|_1+(n\rho/2)\|\bar{y}-\bar{x}^{k+1}-(1/\rho)\lambda^k\|_2^2\right\} \\ \lambda^{k+1} &= \lambda^k+\rho(\bar{x}^{k+1}-\bar{y}^{k+1}) \end{aligned} \tag{5.15}$$

where $f_i(x_i) = (x_i-\theta_i)^T\Upsilon_i(x_i-\theta_i)$, $\bar{x}^k = (1/n)\sum_{i=1}^n x_i^k$, $\bar{y}^k = (1/n)\sum_{i=1}^n y_i^k$, and $z_i^k = x_i^k-\bar{x}^k+\bar{y}^k-(1/\rho)\lambda^k$.

Given that the functions f_i and the l_1 norm are strongly convex, the ADMM updates for x_i^{k+1} and $\bar{y}^{k+1}$ provide solutions to unconstrained convex optimization problems. Consequently, these problems can be solved by equating the derivatives of the objective functions in (5.15) to zero. Subsequently, the closed-form solution for x_i^{k+1} is given by

$$x_i^{k+1} = (2\Upsilon_i+\rho I_p)^{-1}(2\Upsilon_i\theta_i+\rho(x_i^k-\bar{x}^k+\bar{y}^k)-\lambda^k), \tag{5.16}$$

where I_p is the $p \times p$ identity matrix. Similarly, the update for $\bar{y}$ can be expressed as

$$\bar{y}^{k+1} = \begin{cases} (\bar{x}^{k+1} + \lambda^k/\rho) - \frac{\zeta}{\rho}, & \text{if } \bar{x}^{k+1} + \lambda^k/\rho > \frac{\zeta}{\rho} \\ 0, & \text{if } |\bar{x}^{k+1} + \lambda^k/\rho| \leq \frac{\zeta}{\rho} \\ (\bar{x}^{k+1} + \lambda^k/\rho) + \frac{\zeta}{\rho}, & \text{if } \bar{x}^{k+1} + \lambda^k/\rho < -\frac{\zeta}{\rho}. \end{cases} \tag{5.17}$$

5.5.2. Numerical Experiments

We consider the case where $n = 10$. The problem described in (5.14) is solved with three different methods:

1. *Sync*: this algorithm uses ADMM with proximal operator as in (2.15), which simplifies to (2.16) and (2.17) with $\rho = 10$.

2. *STEP-GP*: the algorithm proposed in [23] combining ADMM with proximal operator with GP regression.

3. *STEP-LGP*: the hybrid algorithm proposed in Chapter 2, which combines the regression algorithm developed in Section 2.4.2, the LMMSE approximation presented in Section 2.4.3, and uses an adaptive quantization scheme.

For each of the above algorithms, different quantization methods, or no quantization at all, or censoring methods are considered as follows:

- *Exact*: this method does not employ any quantization, but uses 64-bit floating point numbers.

- *UniQuant*: this is the adaptive uniform quantization method presented in Section 2.4.1 and performed element-wise following the *Uncorrelated Adaptive Scheme* as presented in Section 2.3.2.1. This scheme adapts the middle point and windows length of the quantizer to the conditional mean and covariance matrix given by the regression process, respectively, at each iteration.

- *UniAd-Joint*: The joint adaptive quantization scheme presented in this chapter, which makes its querying decision and quantization resolution allocation as explained in Section 5.3.

- *UniAd-Indiv*: The individual adaptive quantization scheme presented in this chapter, which makes its querying decision and quantization resolution allocation as

Algorithm 6 COCA: Communication-Censored ADMM for the Sharing Problem

Require: $x_i^0 \in \mathbb{R}^p$, $\bar{y}^0 \in \mathbb{R}^p$, $u^0 \in \mathbb{R}^p$, $c \in \mathbb{N}$, $\alpha \in [0,1]$, $w > 0$

for $k = 0, 1, \ldots, k_{\text{stop}}$ **do**
  $\bar{y}^{k+1} \leftarrow \arg\min_{\bar{y} \in \mathbb{R}^p} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^k - u^k\|^2 \right\}$
  for each agent i **do**
    $\nabla \tilde{f}_i^{1/\rho}(z_i^{k-1})$ is the previously transmitted Moreau Envelope gradient before iteration k.
    Coordinator sends $z_i^k \leftarrow x_i^k - \bar{x}^k + \bar{y}^{k+1} - u^k$
    Compute $f_i^{1/\rho}(z_i^k)$ and $\nabla f_i^{1/\rho}(z_i^k)$
    Calculate $\xi_i^k = \nabla f_i^{1/\rho}(z_i^k) - \nabla \tilde{f}_i^{1/\rho}(z_i^{k-1})$
    Calculate $H_i(k, \xi_i^k) = \|\xi_i^k\|_2 - w\alpha^k$
    if $H_i(k, \xi_i^k) \geq 0$ **then**
      Send $\nabla f_i^{1/\rho}(z_i^k)$ to Coordinator
      $x_i^{k+1} \leftarrow z_i^k - (1/\rho)\nabla f_i^{1/\rho}(z_i^k)$
    else
      $x_i^{k+1} \leftarrow z_i^k - (1/\rho)\nabla \tilde{f}_i^{1/\rho}(z_i^k)$
    end if
  end for
  $\bar{x}^{k+1} \leftarrow (1/n)\sum_{i=1}^n x_i^{k+1}$
  $u^{k+1} \leftarrow u^k + \bar{x}^{k+1} - \bar{y}^{k+1}$
  If $\|\bar{x}^k - \bar{y}^k\|_\infty \leq \epsilon_p(1 + \|\lambda^k/\rho\|_\infty)$ then Terminate.
end for

explained in Section 5.4.

- *COCA*: The censoring method presented in [49]. This method checks if there is a considerable variation between the current agent's response and the previously transmitted one. If there is not enough variation, the coordinator uses the previously transmitted reply in this iteration for the ADMM updates. The specifics of this censoring method in the context of our problem are presented in Algorithm 6.

- *QuantRef*: The quantization refinement scheme presented in [50]. This simple quantization scheme sets the middle point to the previous quantized agent's response and adapts the windows length by making it decay at each iteration. The specifics of this adaptive quantization in the context of our problem are presented in Algorithm 7.

In our results, we consider the following combinations: *STEP-GP:Exact*, *STEP-LGP:UniQuant*, *STEP-LGP:UniAd-Joint*, *STEP-LGP:UniAd-Indiv*, *Sync:COCA*, and *Sync:QuantRef*.

Algorithm 7 Quantization Refinement Scheme

Require: $x_i^0 \in \mathbb{R}^p$, $\bar{y}^0 \in \mathbb{R}^p$, $u^0 \in \mathbb{R}^p$, $c \in \mathbb{N}$, $b \in \mathbb{N}$, $\alpha \in [0,1]$

for $k = 0, 1, \ldots, k_{\text{stop}}$ **do**

  $\bar{y}^{k+1} \leftarrow \arg\min_{\bar{y} \in \mathbb{R}^p} \left\{ h(n\bar{y}) + (n\rho/2)\|\bar{y} - \bar{x}^k - u^k\|^2 \right\}$

  for each agent i **do**

    $z_i^k \leftarrow x_i^k - \bar{x}^k + \bar{y}^{k+1} - u^k$

    Send z_i^k to Agent i

    Compute $f_i^{1/\rho}(z_i^k)$ and $\nabla f_i^{1/\rho}(z_i^k)$ ▷ Agent i

    Update $l_i^{k+1} = c\alpha^k$ ▷ Agent i

    Calculate $\nabla \hat{f}_i^{1/\rho}(z_i^k) = \mathbb{Q}(\nabla f_i^{1/\rho}(z_i^k); \nabla \hat{f}_i^{1/\rho}(z_i^{k-1}), l_i^k, b)$ ▷ Agent i

    Send $\nabla \hat{f}_i^{1/\rho}(z_i^k)$ to coordinator. ▷ Agent i

    $x_i^{k+1} \leftarrow z_i^k - (1/\rho)\nabla \hat{f}_i^{1/\rho}(z_i^k)$

  end for

  $\bar{x}^{k+1} \leftarrow (1/n)\sum_{i=1}^n x_i^{k+1}$

  $u^{k+1} \leftarrow u^k + \bar{x}^{k+1} - \bar{y}^{k+1}$

  If $\|\bar{x}^k - \bar{y}^k\|_\infty \leq \epsilon_p(1 + \|\lambda^k/\rho\|_\infty)$ then Terminate.

end for

The numerical experiments were implemented in MATLAB. The solutions of the minimization problems (5.14) are obtained directly using a convex solver from the YALMIP toolbox [41]. We used the GPstuff toolbox [42] for the regression training and inference.

5.5.3. Metrics and Considerations

5.5.3.1. Communication Metric

A communication cost metric can be derived following the contention tree algorithm and the derivations presented in [51]. The contention tree algorithm defines t contending transmitters that want to transmit to an m number of slots. The number of frames is $L_{t,m}$, the number of slots is $mL_{t,m}$, and the number of bits in the payload for agent i is $B_i^k mL_{t,m}$, where B_i^k is the number of bits per slot of agent i. The work in [51] presents statistical results for the contention tree algorithm and presents the expectation

of the variable L_t in terms of t and m, given by

$$\bar{L}_{t,m} \simeq \frac{t}{\log m}$$

In the context of our problem, the agents transmit their Moreau Envelope which is a scalar, and its gradient with dimension p, so each agent transmits a variable with dimension $p+1$. So, the number of bits per slot is

$$B_i^k = (p+1)b_i^k$$

Following the proposed adaptive quantization scheme, the value of b_i^k varies from agent to agent then the metric would be expressed as:

$$TBits = (p+1)\sum_{j=1}^{k_c}\left\lceil \frac{1}{\|\gamma^j\|_1}\sum_{i=1}^{n} b_i^j \right\rceil m^j \bar{L}_{\|\gamma^j\|_1},$$

where k_c is the iteration where convergence was reached, $L_{\|\gamma^k\|_1}$ accounts for the number of frames.

5.5.3.2. ADMM Termination Criterion

For our numerical experiments, we use the ADMM termination criterion presented in Section 4.6.4.2 in Chapter 4.

5.5.3.3. Performance Trade-off

We propose to present the results showing directly the trade-off between the total transmitted bits and the accuracy of the algorithm. Define the negative logarithm of the relative error ($NLRE$) expression as

$$NLRE = -\log(|J_{gt} - J_*|/J_{gt}), \tag{5.18}$$

where J_{gt} is the true optimal value calculated directly with a convex solver, and J_* is the objective value obtained by a particular approach. Also, let us define the logarithm of the

total number of bits (LTbits) transmitted as

$$\text{LTBits} = \log\left((p+1)\sum_{j=1}^{k_c}\left\lceil\frac{1}{\|\gamma^j\|_1}\sum_{i=1}^{n}b_i^j\right\rceil m^j\bar{L}_{\|\gamma^j\|_1}\right). \tag{5.19}$$

We present our results in a graph where the vertical axis shows the values of LT-Bits and the horizontal axis shows the values of $NLRE$. Each point in the graph is a tuple of total transmitted bits and accuracy, and its location shows how well it performs in terms of the trade-off between these two metrics. In particular, the ideal scenario is when $NLRE$ is as large as possible and LTbits is as small as possible. Hence, we want the points to be as close to the right lower corner of the graph as possible.

5.5.4. Initial Parameter Tuning

Since the initial threshold and decay rate variation affect the tested algorithms' overall performance, we propose fine-tuning these parameters for the multiple methods proposed in this work. The threshold mechanism presented in Section 5.3.3 initializes its initial threshold ψ^{k_0} following the expression in (5.12). This initialization requires manually setting the variables ι and ι_Q, which indicate how proportional we want V^{k_0} and $V^{Q[k_0]}$ to be with respect to ψ^{k_0} and $\psi^{Q[k]}$. For the *STEP-LGP:UniAd-Joint* and *STEP-LGP:UniAd-Indiv* methods presented in this chapter, we tune ψ^{k_0} considering ι and ι_Q in the range $[0.7, 0.8\ldots, 1.2, 1.3]$. In addition, we consider the decay rate $\alpha = [0.85, 0.86\ldots, 0.94, 0.95]$ and the quantization decay rate $\alpha_Q = [0.65, 0.66\ldots, 0.74, 0.75]$.

The algorithms used for comparison also have initial parameters that require tuning. For *STEP-GP:Exact* and *STEP-LGP:UniQuant*, we consider the same variation of the initial threshold and decay rate as ι and α considered for *STEP-LGP:UniAd-Joint* and *STEP-LGP:UniAd-Indiv*. The *STEP-LGP:UniQuant* method does not adapt its quantiza-

tion resolution, so for all tested results for this algorithm we fixed the resolution to 9 bits. With respect to *Sync:COCA*, it tunes the constant that multiplies the decay rate (w on line 8 of Algorithm 6) by assigning the values $[1, 1.5, 2, 2.5, 3]$ and setting the decay rate in the range $[0.81, 0.82, \ldots, 0.87]$. Finally, *Sync:QuantRef* tunes its quantization resolution from 8 to 14 bits, the constant that multiplies the decay rate in the range $[1.5, 2, 2.5, 3]$, and the decay rate in the range $[0.9, 0.91, \ldots, 0.99]$.

5.5.5. Numerical Experiment Results

In this subsection, we present the results for 10 agents when the dimension of the variables is set to $p = 5$. We consider tuning the initial parameters of all tested methods following the description in Section 5.5.4. Each graph presented shows results for different sets of M_i, M_h, w_i, w_h, c_i and c_h. In the generated graphs, each point among the same colored cluster represents a ranked tuple of metrics $NLRE$ and LTBits, as presented in Section 5.5.3.3. This ranking is done by setting a tuple as an upper bound with a value of $NLRE$ and LTBits that is higher than any of the values obtained in our results. Then we will calculate the Euclidean distance of all the points obtained across the different initial parameters considered to the upper bound tuple. The 11 points that reach the lowest distance are included in the graph. This set of results considered values of $\eta = 0.2$, $\epsilon_d = 1$, $\rho = 10$, $p = 5$, an absolute tolerance value of $\epsilon^{\text{abs}} = 10^{-6}$, a relative tolerance value of $\epsilon^{\text{rel}} = 10^{-5}$, $x_i^0 = \bar{y}^0 = u^0 = 0$, and constant $c = 3$ for quantization. Also, we consider a minimum quantization resolution of 8 bits and a maximum resolution of 14 for all the methods that requires quantization.

In Figure 5.2 we present the tradeoff performance of all algorithms considered

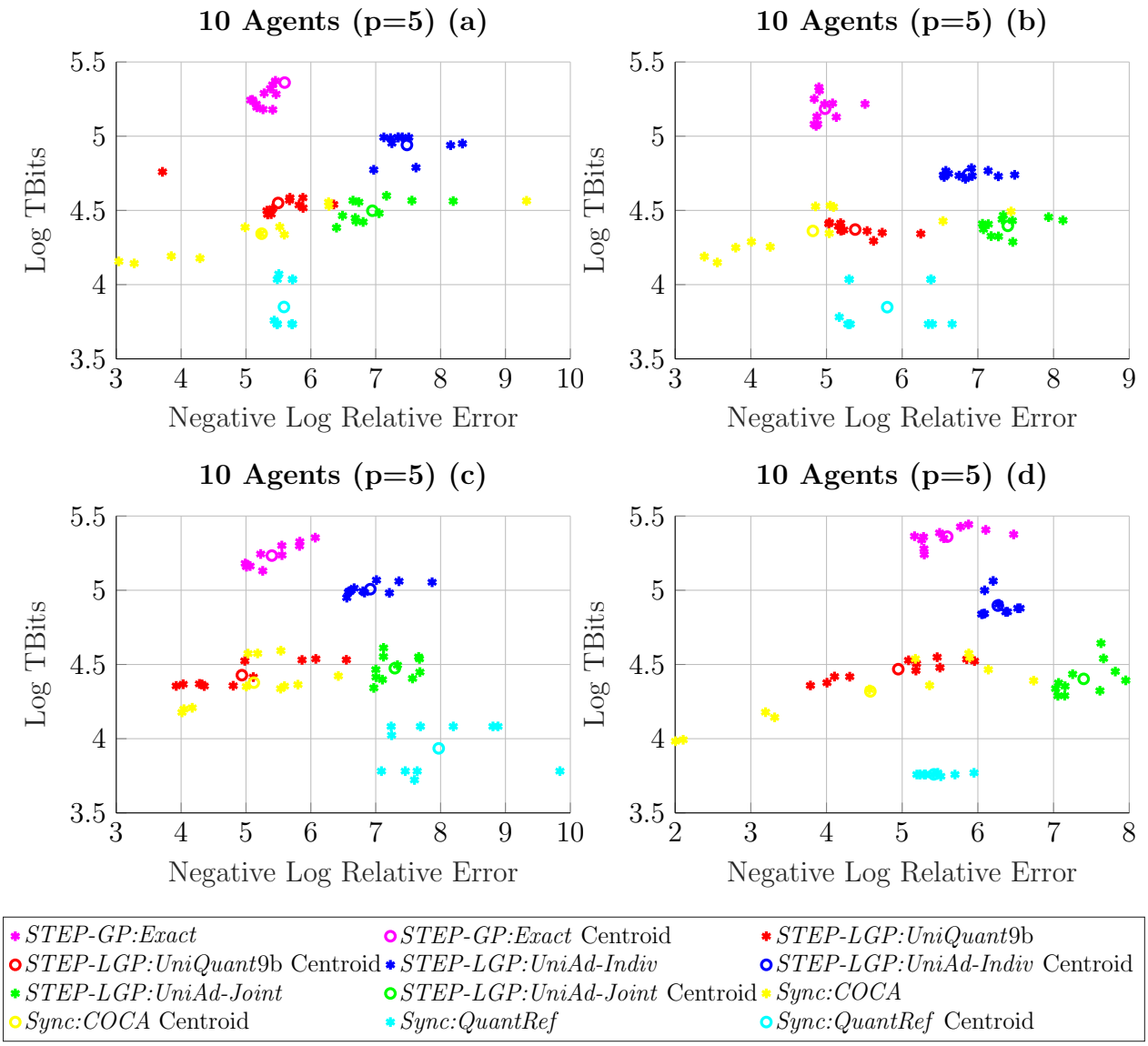

Figure 5.2. Performance trade-off between the Logarithm of the Total Transmitted Bits and the Negative Logarithm of the Relative Error for 10 Agents with variable's dimension $p = 5$. The plots show the 11 best-ranked tuples for four different sets of parameters M_i, M_h, w_i, w_h, c_i, and c_h.

for 10 Agents with variable's dimension $p = 5$. The plots show the 11 best-ranked tuples for four different sets of parameters M_i, M_h, w_i, w_h, c_i, and c_h. The different plots also present the centroid among ranked tuples of the same color. It can be observed that among all the scenarios tested, *STEP-GP:Exact* presents the worst tradeoff between accuracy and total transmitted bits. In addition, *STEP-LGP:UniQuant* is as good in commu-

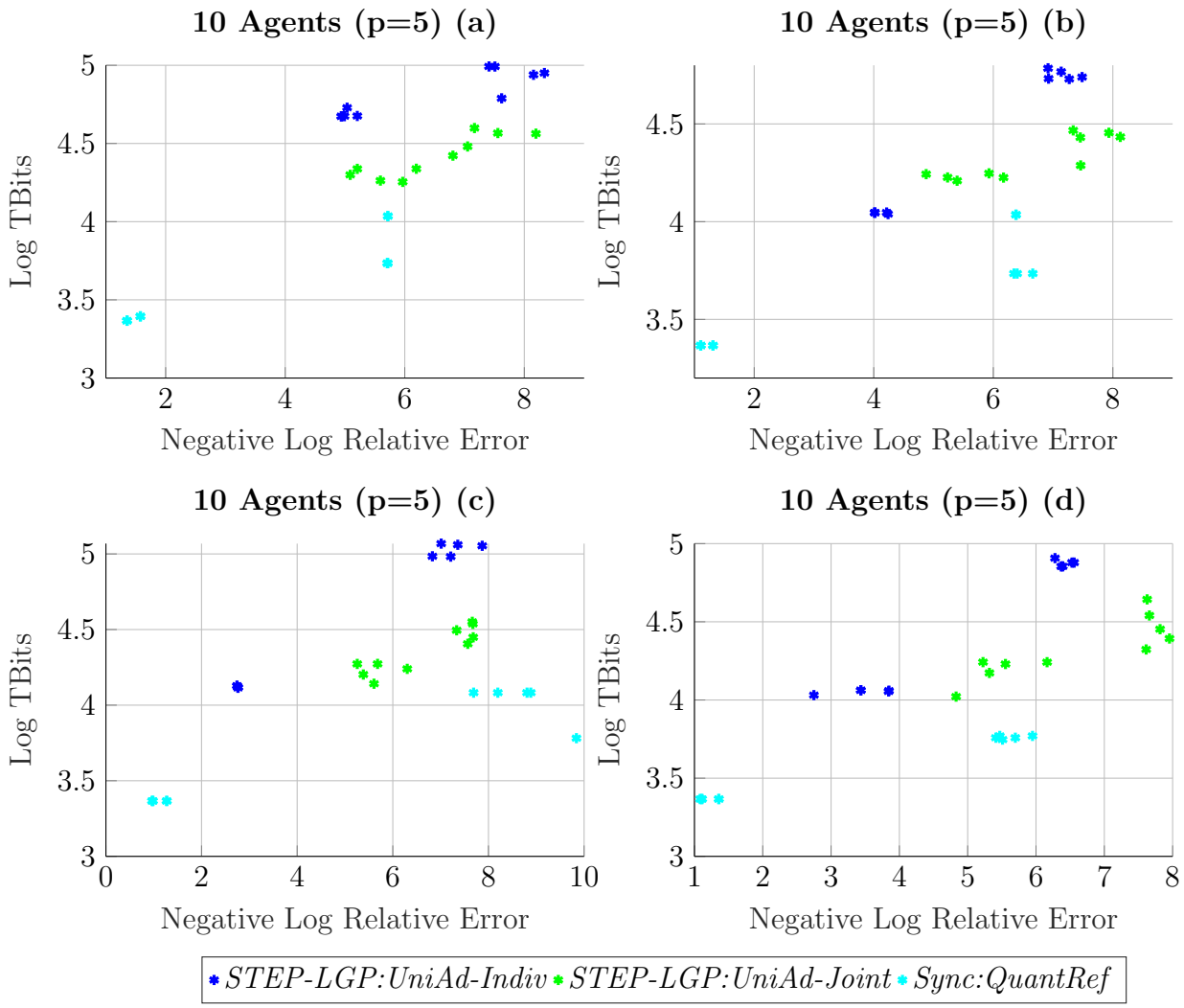

Figure 5.3. Top 5 best results in terms of the Logarithm of the Total Transmitted Bits and Top 5 best results in terms of Negative Logarithm of the Relative Error for 10 Agents with variable's dimension $p = 5$ for *STEP-LGP:UniAd-Joint*, *STEP-LGP:UniAd-Indiv*, and *Sync:QuantRef*. The plots are generated for four different sets of parameters M_i, M_h, w_i, w_h, c_i, and c_h.

nication reduction as *Sync:COCA* and *STEP-LGP:UniAd-Joint*, but in all cases it is close to the worst Negative Logarithm of the Relative Error values. Although some points of *Sync:COCA* compete among the best trade-off points as in Figures 5.2 (a) and (b), it also presents the worst overall accuracy values in all cases. This shows that *Sync:COCA* is susceptible to initial parameter tuning, allowing it to achieve remarkable results if tuned cor-

rectly, but also presents a very poor accuracy otherwise. In all cases, *STEP-LGP:UniAd-Indiv* presents a good Negative Logarithm of the Relative Error values but presents the most transmitted bits only behind *STEP-GP:Exact*. Finally, the best results are obtained by *STEP-LGP:UniAd-Joint* and *Sync:QuantRefL*. In Figures 5.2 (a),(b) and (d), *STEP-LGP:UniAd-Joint* presents the best accuracy results, while *Sync:QuantRef* presents the most communication reduction. The exception is Figure 5.2 (c) where *Sync:QuantRef* presents the best overall results.

Due to the competing results between *STEP-LGP:UniAd-Joint* and *Sync:QuantRef* we present the results in Figure 5.3. This set of graphs presents the top 5 points in terms of the Logarithm of the Total Transmitted Bits and the top 5 best results in terms of Negative Logarithm of the Relative Error for 10 Agents with variable's dimension $p = 5$ for *STEP-LGP:UniAd-Joint*, *STEP-LGP:UniAd-Indiv*, and *Sync:QuantRef*. These figures use the same sets of problem parameters M_i, M_h, w_i, w_h, c_i, and c_h as in Figure 5.2. The results in Figure 5.3 show a clear trend that the best 5 points in terms of communication and the best 5 points in terms of precision for *STEP-LGP:UniAd-Joint* are significantly closer together than for *STEP-LGP:UniAd-Indiv* and *Sync:QuantRef*. These results show that *STEP-LGP:UniAd-Joint* is the most robust method that presents consistent results regardless of initial parameter tuning. Thus, even though *STEP-LGP:UniAd-Joint* and *Sync:QuantRef* present the best results in terms of accuracy and communication, respectively, it is *STEP-LGP:UniAd-Joint* that is more reliable without regard to the considered initial conditions.

5.6. Conclusion to Chapter 5

In this chapter, we extend the LGP algorithm presented in Chapter 2 to allow adaptive quantization resolution. We proposed two new quantization schemes: one that makes the communication decision and the bits for quantization collectively, and another that does so individually. Numerical solutions to a distributed sharing problem showed that our proposed collective method becomes significantly more accurate than the quantization scheme presented in Chapter 2 while maintaining a similar reduction in communication. On the other hand, the proposed individual adaptive quantization scheme also presented better accuracy compared to the method in Chapter 2 but it presented more communication overhead. Finally, compared to a proposed censoring method and quantization scheme in other works, our proposed collective method was competitive for the best tradeoff between communication reduction and accuracy, while presenting the most robustness against the initial parameters tuning.

Chapter 6. Conclusions and Future Directions

6.1. Conclusions

In distributed optimization frameworks where a cluster of agents interfaces with a central coordinator, the optimization process typically entails each agent tackling individual local subproblems privately while engaging in frequent data exchanges with the coordinator to collectively solve the overarching distributed problem. In such scenarios, the conventional query-response mechanism tends to escalate communication expenses for the system, thereby prompting the need for communication minimization, particularly in situations where communication resources are constrained or expensive. Integrating Gaussian processes (GP) as a learning component to the Alternating Direction Method of Multipliers (ADMM) has proven effective in learning each agent's local proximal operator to reduce the required communication exchange. For this reason, the initial stage of this work (Chapter 2) proposes a novel hybrid method named LGP that integrates GP-based learning with an adaptive uniform quantization strategy to further minimize communication costs in distributed optimization. Quantization is used to reduce the communication overhead even further by reducing the payload of the shared information. Also, this initial proposed quantization scheme sets its middle point and windows length to the conditional mean and covariance given by the regression process, respectively. Although the resulting quantization error deviates from a Gaussian distribution, we introduced a new regression algorithm. Inspired by GP, this algorithm, termed LGP-R, employed a Linear Minimum Mean Square Estimator that factored in the statistics of the quantization error. Furthermore, communication overhead was mitigated by enhancing the uniform quantizer

through an orthogonalization process of its input, addressing inherent input correlation, and incorporating dithering to ensure uncorrelated noise introduction. Through numerical experiments on a distributed sharing problem, our hybrid approaches demonstrated a significant reduction in total communication costs compared to baseline methods, even achieving global solution discovery at low quantization resolutions.

Next, we continue our study by presenting a convergence analysis for the STEP-GP and LGP algorithms. These analyzes were based on the convergence analysis of the generalized ADMM and SI-ADMM algorithms. Following that, we outline the derivation of a convergence proof for the STEP-GP algorithm, where we establish that the expected value of the ADMM residual converges to zero as the algorithmic iterations approach infinity, achieving this convergence at a geometric rate. For the case of the analysis of the LGP algorithm, we reached a similar conclusion; however, we assumed that the coordinator can vary the quantization resolution at each iteration and that it can assign infinitely large bits for quantization. We also present convergence properties in the case where the quantization resolution is upper bounded using the LGP algorithm, leading to the conclusion that the expectation of the ADMM residual is bounded, and such bound is explicitly displayed. Finally, we present an analysis of the connection between the analysis in this chapter and the algorithms defined in Chapter 2. Since the specific query mechanism used in the LGP algorithm in Chapter 2 is different from the one used in the convergence analysis, we established a direct relationship between the two mechanisms, allowing us to determine that the expectation of the ADMM residual is also bounded for the method presented in Chapter 2.

One of the most important aspects for the correct performance of our LGP algo-

rithm is how we determine the agents to be queried in each iteration. Hence, in Chapter 4 we introduced multiple query strategies aimed at determining whether the coordinator should initiate queries to the agents during a specific iteration when executing the *STEP-GP* algorithm, leveraging the concept of a general querying framework. As this decision-making process significantly influences how regression influences the ADMM algorithm, our objective was to focus on ADMM performance in the absence of quantization. Consequently, in the study in Chapter 4, we intend to examine the effects of various query strategies on ADMM without being influenced by potential quantization errors. The proposed general framework addresses a constrained optimization problem by effectively balancing two conflicting objectives: maximizing communication reduction while minimizing error in the final solution. Motivated by this optimization challenge and an alternative representation of the regular ADMM updates that underscores the inherent interdependence among agents, we proposed a collective query strategy to minimize a convex communication cost constrained by the trace of the joint uncertainty of the ADMM variables. In contrast, to alleviate the computational overhead imposed on our algorithm, we introduced individual query strategies for each agent, utilizing individual uncertainty metrics to gauge whether the prediction is sufficiently reliable to forego a communication round. Numerical experiments on a sharing problem with quadratic cost functions revealed different performances of the proposed methodologies concerning the trade-off between communication cost reduction and accuracy loss in solving the optimization problem. It is particularly noteworthy that the proposed collective query method achieves superior trade-off performance compared to the independent query strategies.

Finally, in Chapter 5 we expand the LGP algorithm discussed in Chapter 2 to

incorporate adaptive quantization resolution. We introduce two novel quantization approaches: one that jointly determines the communication decision and the quantization bit allocation, and another that handles these aspects independently. Through numerical experiments involving a distributed sharing problem, we demonstrate that our collective quantization method achieves significantly higher accuracy than the quantization scheme outlined in Chapter 2 while maintaining a similar level of communication reduction. In contrast, the individually adaptive quantization scheme also exhibits improved accuracy compared to the method in Chapter 2, albeit with increased communication overhead. Furthermore, in comparison to a censoring method and a quantization scheme proposed in prior research, our collective approach demonstrates competitiveness in achieving the optimal balance between communication reduction and accuracy, while displaying greater robustness against variations in initial parameter settings.

In general, the different algorithms proposed throughout this study achieved their main objective of reducing the overall communication overhead while maintaining satisfactory accuracy in their global solutions. The good accuracy of the numerical experiments is aligned with the derived convergence analysis, where, for the cases where convergence is not guaranteed, we proved that the overall ADMM residual is bounded by a decaying bound, so we can expect the solutions of our algorithms to be in the vicinity of the real solution.

6.2. Future Directions

In this brief section, we present future directions that can be taken from our research.

6.2.1. Alleviate the Computational Burden Coming from the Regression Process

During the course of collecting numerical results using our various proposed methods, we observed that the greatest computational burden comes from the update of the hyperparameters performed each time the training set of the regression process is updated. This optimization in our algorithm uses the square-exponential covariance function defined as

$$\phi(x_s, x_j) = \sigma_f^2 \exp\left(-\frac{1}{2L_s^2}(x_s - x_j)^2\right),$$

where σ_f^2 is the variance of the signal and L_s is the length scale. These two variables are the hyperparameters that are updated at each iteration.

Observations of the numerical results among all algorithms proposed in this work showed that the hyperparameter behavior is similar in most cases. In the first iterations, the hyperparameters start to increase their values rapidly, while close to convergence, the increment of those values is each time smaller. This trend induces us to question whether it is possible to use the increasing trend of the hyperparameters updates to skip such update procedures in some iterations or skip them completely once we reach the iteration where the hyperparameters values do not vary much. The reduction of the hyperparameter updates would significantly reduce the computation complexity of our proposed algorithms, making them suitable for applications where high computation complexity is undesirable.

6.2.2. Extend the Derived Convergence Analysis for LGP

The convergence analysis presented in Section 3.6 of Chapter 3, presents an analysis for the LGP algorithm when the quantization resolution is constrained. In this dis-

cussion, we defined k' as the last iteration where the global uncertainty constraint is met before the quantization uncertainty can't decrease further. This convergence analysis would become stronger if some property or bound could be imposed on k'. The challenge is that this involves bounding the sum of the traces of the covariance matrices of the agents, which depends on the iteration k', which is unknown. An in-depth analytical study can be conducted in this matter to achieve a complete and global proof of convergence for the LGP algorithm.

6.2.3. Enhance the Numerical Examples

In this work, we conducted extensive numerical experiments to test and compare the different proposed methods. We considered using an optimization problem involving quadratic functions for its simplicity and properties. It will be beneficial to test our proposed algorithms in more complex problems with clear real-life applicability.

6.2.4. Regression Improvement

When performing the prediction, we could exploit the similarity between local objective functions, which may be captured and characterized by vector Gaussian processes. At the center, we may exploit such correlation across agents to further improve the accuracy of regression using the corresponding vector GP. However, it should be noted that there will be an asymmetry between the models used by the center and agents. How we can resolve such an asymmetry in both regression and optimization problems remains as one of our future problems to tackle.

6.2.5. Better Communication Channel Modeling

In our numerical results, we account for channel contention among simultaneous communications, either through a MAC modeling or using the contention tree algorithm. However, these techniques are simple and do not account for more complex communication events. For example, due to contention, packets might get lost and never reach their destination. Furthermore, these packets could come with a delay of several iterations, arriving later after the coordinator assumed that the information was lost. Mechanisms for adjusting for such events are an interesting topic to study.

Appendix A. Proof of Proposition 1 in Chapter 2

Define $x = y - \mu_y \sim \mathcal{N}\left(0, \sigma_y^2\right)$. The output of the adaptive uniform quantizer is given by the standard uniform quantizer $\mathbb{Q}_{\mathrm{u}}(y; \mu_y, \frac{2c\sigma_y}{2^b})$, which is equivalent to $\mu_y + \mathbb{Q}_{\mathrm{u}}(x; 0, \frac{2c\sigma_y}{2^b})$. Using the result presented in [39, Section V-A] on the quantization error of a uniform quantizer on a zero-mean Gaussian random variable, we can derive the equations of $\mathbb{E}[\epsilon_{\mathbb{Q}}]$ and $\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]$.

The correlation between y and $\epsilon_{\mathbb{Q}}$ is

$$\mathbb{E}[y\epsilon_{\mathbb{Q}}] = \mathbb{E}[(x + \mu_y)\epsilon_{\mathbb{Q}}] = \mathbb{E}[x\epsilon_{\mathbb{Q}}] + \mu_y\mathbb{E}[\epsilon_{\mathbb{Q}}] = \mathbb{E}[x\epsilon_{\mathbb{Q}}].$$

Using the result presented in [39, Section V-B] on the correlation between a zero-mean Gaussian random variable and its uniform quantization error, we have that

$$\mathbb{E}[x\epsilon_{\mathbb{Q}}] = 2\sigma_y \sum_{m=1}^{\infty} (-1)^m \exp\left(-2\pi^2 m^2 r^2\right),$$

which results in the same equation for $\mathbb{E}[y\epsilon_{\mathbb{Q}}]$.

Appendix B. Proof of Lemmas 1 and 2 in Chapter 2

We first need the following result.

Proposition 6 *For* $r > \frac{1}{\sqrt{2\pi}}$,

$$\sum_{m=1}^{\infty}(-1)^m m^2 \exp\left(-2\pi^2 m^2 r^2\right) < 0.$$

Proof: Define $S(m) = m^2 \exp\left(-2\pi^2 m^2 r^2\right)$. Then the series is $\sum_{m=1}^{\infty}(-1)^m S(m)$. We have

$$\begin{aligned}\frac{\mathrm{d}S(m)}{\mathrm{d}m} &= 2m\exp\left(-2\pi^2 m^2 r^2\right) - 4\pi^2 r^2 m^3 \exp\left(-2\pi^2 m^2 r^2\right) \\ &= 2m\exp\left(-2\pi^2 m^2 r^2\right)\left(1 - 2\pi^2 r^2 m^2\right).\end{aligned}$$

For $r > \frac{1}{\sqrt{2\pi}}$ and $m \geq 1$, we have $1 - 2\pi^2 r^2 m^2 < 0$, thus $\frac{\mathrm{d}S(m)}{\mathrm{d}m} < 0$, which implies that $S(m)$ is strictly decreasing with m, i.e., $S(1) > S(2) > S(3) > S(4) > \ldots$. Therefore, the series is $\sum_{m=1}^{\infty}(-1)^m S(m) = (-S(1) + S(2)) + (-S(3) + S(4)) + \cdots < 0$. □

We will now prove Lemmas 1 and 2. Consider the series $s(r) = \sum_{m=1}^{\infty} \frac{(-1)^m}{m^2} \exp\left(-2\pi^2 m^2 r^2\right)$ as a function of r. Define $s_m(r) = \frac{1}{m^2}\exp\left(-2\pi^2 m^2 r^2\right)$. Then $s(r) = \sum_{m=1}^{\infty}(-1)^m s_m(r)$. For an integer $m \geq 1$, we have

$$\begin{aligned}s_{m+1}(r) &= \frac{1}{(m+1)^2}\exp\left(-2\pi^2(m+1)^2 r^2\right) \\ &< \frac{1}{m^2}\exp\left(-2\pi^2(m+1)^2 r^2\right) \\ &= \frac{1}{m^2}\exp\left(-2\pi^2 m^2 r^2\right)\exp\left(-2\pi^2(2m+1)r^2\right) \\ &< \frac{1}{m^2}\exp\left(-2\pi^2 m^2 r^2\right) \\ &= s_m(r),\end{aligned}$$

where the last inequality holds due to $\exp\left(-2\pi^2(2m+1)r^2\right) < 1$. Therefore

$$s(r) = (-s_1(r) + s_2(r)) + (-s_3(r) + s_4(r)) + \cdots < 0.$$

Using the same approach, we can show that $\sum_{m=1}^{\infty}(-1)^m \exp\left(-2\pi^2 m^2 r^2\right) < 0$.

To show that $s(r)$ is increasing with r, we differentiate it with respect to r:

$$\frac{\mathrm{d}s(r)}{\mathrm{d}r} = -4\pi^2 r \sum_{m=1}^{\infty}(-1)^m \exp\left(-2\pi^2 m^2 r^2\right)$$

which is positive because we have just shown that $\sum_{m=1}^{\infty}(-1)^m \exp\left(-2\pi^2 m^2 r^2\right) < 0$. Therefore, $s(r)$ is increasing with r.

Similarly, for the series in Lemma 2, we have

$$\frac{\mathrm{d}}{\mathrm{d}r}\sum_{m=1}^{\infty}(-1)^m \exp\left(-2\pi^2 m^2 r^2\right) = -4\pi^2 r \sum_{m=1}^{\infty}(-1)^m m^2 \exp\left(-2\pi^2 m^2 r^2\right) > 0$$

for all $r > \frac{1}{\sqrt{2\pi}}$, due to Proposition 6. Therefore, the series $\sum_{m=1}^{\infty}(-1)^m \exp\left(-2\pi^2 m^2 r^2\right)$ is increasing with r for all $r > \frac{1}{\sqrt{2\pi}}$.

Appendix C. Proof of Proposition 3 in Chapter 2

The dequantized value $\hat{y}$ will be $\hat{y} = A^{-1}\mathbb{Q}_{\text{ua}}(y^A; 0, \sigma_w, c, b) + \mu_y$, but can be also expressed as

$$\hat{y} = A^{-1}[A(y - \mu_y) + \epsilon_{\mathbb{Q}}] + \mu_y = y + A^{-1}\epsilon_{\mathbb{Q}} = y + \hat{\epsilon}_{\mathbb{Q}}.$$

Analyzing the auto correlation of $\hat{\epsilon}_{\mathbb{Q}}$ we have:

$$\begin{aligned}\mathbb{E}[\hat{\epsilon}_{\mathbb{Q}}\hat{\epsilon}'_{\mathbb{Q}}] &= (A)^{-1}\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]((A)^{-1})' \\ &= (A)^{-1}\Lambda_{\epsilon_{\mathbb{Q}}}((A)^{-1})',\end{aligned}$$

where $\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}]$ is the auto correlation of the quantization error and $\Lambda_{\epsilon_{\mathbb{Q}}}$ is a diagonal matrix with its diagonal given by the vector$\frac{v(2^b/2c)}{12}\tilde{q}^2$, with $v(2^b/2c)$ as defined in Proposition 1.

If A_1 is used then $\tilde{q}$ will be $\tilde{q} = \frac{2c}{2^b}I_{p+1} = \Gamma(b,c)I_{p+1}$, where $\Gamma(b,c) = \frac{2c}{2^b}$. On the other hand, if A_2 is used then $\tilde{q} = \frac{2c}{2^b}\sqrt{\Lambda} = \Gamma(b,c)\sqrt{\Lambda}$. Therefore we will have that

$$\begin{aligned}\mathbb{E}[\hat{\epsilon}_{\mathbb{Q}}\hat{\epsilon}'_{\mathbb{Q}}] &= A^{-1}\Lambda_{\epsilon_{\mathbb{Q}}}(A^{-1}) \\ &= \frac{\Gamma^2(b,c)v(2^b/2c)}{12}(A^{-1}\tilde{\Lambda}_{\epsilon_{\mathbb{Q}}}(A^{-1})'),\end{aligned}$$

with $\tilde{\Lambda}_{\epsilon_{\mathbb{Q}}}$ being I_{p+1} or Λ depending on the selection of A. Finally, we have that since $A^{-1}\tilde{\Lambda}_{\epsilon_{\mathbb{Q}}}(A^{-1})' = \Sigma_y$, then no matter the selection of A the result will be

$$\mathbb{E}[\hat{\epsilon}_{\mathbb{Q}}\hat{\epsilon}'_{\mathbb{Q}}] = \frac{\Gamma^2(b,c)v(2^b/2c)}{12}\Sigma_y = \Delta.$$

Appendix D. Proof of Theorem 1 in Chapter 2

The proposed LMMSE will be given by the linear combination

$$\mu(x_*) = H\hat{Y}. \tag{D.1}$$

Then, if (D.1) is a LMMSE then it must follow the orthogonal principle which will be given by $\mathbb{E}[(\mu(x_*) - \hat{y}_*)(\hat{Y})'] = 0$. From this point we can obtain an expression for H

$$\mathbb{E}[(H\hat{Y} - \hat{y}_*)(\hat{Y})'] = 0$$

$$H\mathbb{E}[(Y + \epsilon_n + \epsilon_{\mathbb{Q}})(Y + \epsilon_n + \epsilon_{\mathbb{Q}})'] = \Phi(x_*, X). \tag{D.2}$$

Since ϵ_n is independent from the rest, all cross products involving ϵ_n will be turn to zero by the expectation. Therefore we can simplify the expression to

$$H[\Phi(X, X) + \mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}] + \sigma_n I_{m(p+1)} + 2\mathbb{E}[Y\epsilon'_{\mathbb{Q}}] = \Phi(x_*, X). \tag{D.3}$$

Defining $\mathbb{E}[\epsilon_{\mathbb{Q}}\epsilon'_{\mathbb{Q}}] = \Delta$, we have the expression

$$H = \Phi(x_*, X)[\Phi(X, X) + \Delta + \sigma_n I_{m(p+1)} + 2\mathbb{E}[Y\epsilon'_{\mathbb{Q}}]]^{-1}. \tag{D.4}$$

The term $\mathbb{E}[Y\epsilon'_{\mathbb{Q}}]$ expresses the correlation between the input of the quantizer and the quantization error. In Proposition 1 a way to calculate this correlation is presented. Because we subtract the mean of the input of the quantizer before performing the quantization, we have $\mathbb{E}[\epsilon'_{\mathbb{Q}}] = 0$, following Proposition 1. Thus, the following holds true, $\mathbb{E}[Y\epsilon'_{\mathbb{Q}}] = \mathbb{E}[(Y - \mu(Y))\epsilon'_{\mathbb{Q}}] + \mu(Y)\mathbb{E}[\epsilon'_{\mathbb{Q}}] = \mathbb{E}[(Y - \mu(Y))\epsilon'_{\mathbb{Q}}]$. This means that the results of Proposition 1 can be extended to calculate the elements conforming matrix $\mathbb{E}[Y\epsilon'_{\mathbb{Q}}]$. This is done directly for the diagonal terms that come from the same dimension, for example, $\mathbb{E}[Y_{[1]}\epsilon'_{\mathbb{Q}[1]}]$ where $Y_{[1]}$ and $\epsilon'_{\mathbb{Q}[1]}$ refer to the first element of vectors Y and $\epsilon'_{\mathbb{Q}}$, respectively.

In case we want to calculate $\mathbb{E}[Y_{[i]}\epsilon'_{\mathbb{Q}[j]}]$, $i \neq j$, we define $\tilde{Y}_{[i]} = Y_{[i]} - \mu(Y_{[i]})$ and do the following:

$$\mathbb{E}[Y_{[i]}\epsilon'_{\mathbb{Q}[j]}] = \mathbb{E}[\tilde{Y}_{[i]}\epsilon'_{\mathbb{Q}[j]}] = \mathbb{E}[(\tilde{Y}_{[i]} - \xi_{ij}\tilde{Y}_{[j]} + \xi_{ij}\tilde{Y}_{[j]})\epsilon'_{\mathbb{Q}[j]}]$$
$$= \mathbb{E}[(\tilde{Y}_{[i]} - \xi_{ij}\tilde{Y}_{[j]})\epsilon'_{\mathbb{Q}[j]}] + \xi_{ij}\mathbb{E}[\tilde{Y}_{[j]}\epsilon'_{\mathbb{Q}[j]}],$$

where $\xi_{ij}\tilde{Y}_{[j]}$ is the MMSE of $\tilde{Y}_{[i]}$ with ξ_{ij} being the operator to estimate $\tilde{Y}_{[i]}$ from $\tilde{Y}_{[j]}$. Since the error of the MMSE is given by $\epsilon_{ij} = \tilde{Y}_{[i]} - \xi_{ij}\tilde{Y}_{[j]}$, then ϵ_{ij} is independent of $\tilde{Y}_{[j]}$. Therefore,

$$\mathbb{E}[(\tilde{Y}_{[i]} - \xi_{ij}\tilde{Y}_{[j]})\epsilon'_{\mathbb{Q}[j]}] = \mathbb{E}[\tilde{Y}_{[i]} - \xi_{ij}\tilde{Y}_{[j]}]\mathbb{E}[\epsilon'_{\mathbb{Q}[j]}] = 0.$$

Thus,

$$\mathbb{E}[Y_{[i]}\epsilon'_{\mathbb{Q}[j]}] = \xi_{ij}\mathbb{E}[\tilde{Y}_{[j]}\epsilon'_{\mathbb{Q}[j]}].$$

Consequently, we can calculate any correlation $\mathbb{E}[Y_{[i]}\epsilon'_{\mathbb{Q}[j]}]$ following the correlation expression presented in Proposition 1.

Finally, the error covariance of the estimator will be given by

$$\Sigma(x_*) = \mathbb{E}[(\hat{y}_* - H\hat{Y})(\hat{y}_* - H\hat{Y})^T].$$

Expanding this expression and operating the expectations we get

$$\Sigma(x_*) = \Phi(X_*, X_*) - H^T\Phi(X, X_*) - \Phi(X_*, X)H - H^T\Phi(X, X)H. \tag{D.5}$$

Finally, introducing the expression of H in (D.4) we get

$$\Sigma(x_*) = \Phi(X_*, X_*) - \Phi(X_*, X)[\Phi(X, X) + \sigma_n^2 I_{m(p+1)} + \Delta + 2\mathbb{E}[Y\epsilon'_{\mathbb{Q}}]]^{-1}\Phi(X, X_*).$$

Appendix E. Proof of Theorem 2 in Chapter 2

The expression for our estimator will be defined as

$$\bar{y}_* - \mu(x_*) = B(\hat{y}_* - \mu(x_*)),$$

where B is the matrix determined by resorting to the orthogonal principle. Using the orthogonal principle for this LMMSE like in the LGP case the expression for B will be

$$B\ E[(\hat{y}_* - \mu(x_*))(\hat{y}_* - \mu(x_*))'] = \mathbb{E}[(\hat{y}_* - \mu(x_*))(\hat{y}_* - \mu(x_*))']. \tag{E.1}$$

So, inserting the definition of $\mu(x_*)$ and $\Sigma(x_*)$ from Theorem 2 into (E.1) will lead to the simplified version

$$B = \Sigma(x_*)[\Sigma(x_*) + \sigma_n I_{p+1} + \Delta_{p+1} + 2\mathbb{E}[y_* \epsilon'_{\mathbb{Q}*}]]^{-1}$$

Appendix F. Details of MAC Metric

Assuming that the coordinator communicates with the agents wirelessly following the IEEE 802.11 specification, a MAC layer simulator was implemented. The 802.11 CSMA/CA simulator presented in [43] was chosen because of its simplicity, which was modified to our purposes. The simulator implemented in MATLAB will return the number of total transmissions, successful transmissions, and an efficiency value defined by $\xi = st/tt$, where st is the successful transmissions observed and tt the total amount of transmissions performed. The simulation was run offline 1000 times to obtain an average efficiency ξ. Once the average values are obtained for different payloads and number of agents, those values will be used with the results given by the distributed optimization simulation to calculate the communication time for each round. In particular, at the k-th iteration, the coordinator will receive a certain amount of simultaneous responses which are expressed in the variable T^k_{simul}. The expected transmission time in one iteration round will be $T^k_{\text{round}} = T^k_{\text{simul}}/\xi^*$, where ξ^* is the average efficiency in the MAC simulation for the given scenario. The total transmission time will be $Tx_t = \sum_{k=1}^{N} T^k_{\text{round}}$, where N is the number of iterations taken to reach convergence. This metric is not only affected by the total number of communications that were performed but also the number of agents communicating at each iteration and the payload size, thereby making it a more robust metric to compare the performance of the proposed methods.

Appendix G. Proof of Proposition 4 in Chapter 4

Consider the condition in (4.10). We introduce a unitary transformation U, whose columns are normalized eigenvectors of Σ_F, i.e., $\Sigma_F = U\Lambda U^\top$, where Λ is the diagonal matrix whose diagonal entries are the eigenvalues of Σ_F sorted in descending order $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_p > 0$. Given $F \sim \mathcal{N}(\mu, \Sigma_F)$, define $G = U^\top(F - \mu)$, which follows $\mathcal{N}(0, \Lambda)$. Moreover, $\|G\|_2 = \|F - \mu\|_2$. Consequently,

$$\mathrm{P}\big[\|F - \mu\|_2 \leq \|\mu\|_2\delta\big] = \mathrm{P}\big[\|G\|_2 \leq \|\mu\|_2\delta\big] \geq 1 - \xi. \tag{G.1}$$

Let us define $Z_l = \frac{G_l}{\sqrt{\lambda_l}}$ for $1 \leq l \leq p$, with $Z_l \sim \mathcal{N}(0, 1)$. Then, (G.1) can be expressed in terms of Z as

$$\mathrm{P}\left[\sum_{l=1}^{p} \lambda_l Z_l^2 \geq \|\mu\|_2^2\delta^2\right] \leq \xi, \tag{G.2}$$

requiring the probability of being outside of an error sphere to be small.

Let $R = \sum_{l=1}^{p} \lambda_l Z_l^2$, which follows a weighted chi-square distribution, and $X = R - \sum_{l=1}^{p} \lambda_l$, we transform (G.2) as

$$\mathrm{P}\left[X + \sum_{l=1}^{p} \lambda_l \geq \|\mu\|_2^2\delta^2\right] \leq \xi. \tag{G.3}$$

We will follow the proof of Lemma 1 in [52] to get a bound for the inequality in (G.3). For a random vector Z with individual components $Z_l \sim \mathcal{N}(0, 1)$, the logarithm of the Laplace transform of $Z_l^2 - 1$ is given by

$$\psi(u) = \log[\mathbb{E}[\exp(u(Z_l^2 - 1))]] = -u - \frac{1}{2}\log(1 - 2u),$$

which for $0 < u < 1/2$ we get the bound

$$\psi(u) \leq \frac{u^2}{1 - 2u}.$$

Therefore, extending the previous expressions for a variable $Y = \sum_{l=1}^{p} a_l(Z_l^2 - 1)$, with $a_l \geq 0$, we get

$$\log[\mathbb{E}[\exp(uY)]] = \sum_{l=1}^{p} \log\left[\mathbb{E}[\exp(ua_l(Z_l^2 - 1))]\right] \leq \sum_{l=1}^{p} \frac{a_l^2 u^2}{1 - 2a_l u}, \tag{G.4}$$

which leads to the inequality

$$\log[\mathbb{E}[\exp(uY)]] \leq \frac{\|a\|_2^2 u^2}{1 - 2\|a\|_\infty u}. \tag{G.5}$$

On the other hand, in [53] it was proven that if

$$\log[\mathbb{E}[\exp(uY)]] \leq \frac{vu^2}{2(1 - 2cu)}, \tag{G.6}$$

then, for any positive x,

$$\mathrm{P}(Y \geq cx + \sqrt{2vx}) \leq \exp(-x). \tag{G.7}$$

Thus, given (G.5) and (G.6) we get $v/2 = \|a\|_2^2$ and $c = 2\|a\|_\infty$, which allow us to rewrite (G.7) as

$$\mathrm{P}(Y \geq 2\|a\|_\infty x + 2\|a\|_2\sqrt{x}) \leq \exp(-x). \tag{G.8}$$

We can define $\alpha = 2\|a\|_\infty$ and $\beta = 2\|a\|_2$, and by equalling $2\|a\|_\infty x + 2\|a\|_2\sqrt{x}$ to a positive number w we get

$$\alpha x + \beta\sqrt{x} = w$$

$$\alpha x + \beta\sqrt{x} - w = 0.$$

Solving the quadratic equation we get that

$$\sqrt{x} = \frac{-\beta + \sqrt{\beta^2 + 4\alpha w}}{2\alpha},$$

where we can obtain a value for x that depends on w and will be named $x_{(w)}$ defined as

$$x_{(w)} = \frac{\beta^2}{2\alpha^2} - \frac{\beta}{2\alpha^2}\sqrt{\beta^2 + 4\alpha w} + \frac{w}{\alpha}. \tag{G.9}$$

Introducing the definition of α and β into (G.9) we get

$$x_{(w)} = \frac{\|a\|_2^2}{2\|a\|_\infty^2} - \frac{\|a\|_2^2}{2\|a\|_\infty^2}\sqrt{1 + \frac{2w\|a\|_\infty}{\|a\|_2^2}} + \frac{w}{2\|a\|_\infty}, \tag{G.10}$$

which after some algebraic manipulations can be expressed as

$$x_{(w)} = \left(\sqrt{\frac{w}{2\|a\|_\infty} + \frac{\|a\|_2^2}{4\|a\|_\infty^2}} - \frac{\|a\|_2}{2\|a\|_\infty}\right)^2. \tag{G.11}$$

Inserting (G.11) and $\alpha x + \beta\sqrt{x} = w$ into (G.8), we get the expression for the desired probability as

$$\mathrm{P}\left[Y \geq w\right] \leq \exp(-x_{(w)}), \forall w \geq 0. \tag{G.12}$$

Going back to the context of the inequality in (G.3) given by

$$\mathrm{P}\left[X + \sum_{l=1}^{p} \lambda_l \geq \|\mu\|_2^2\delta^2\right] \leq \xi,$$

and since $\sum_{l=1}^{p} \lambda_l = \mathrm{tr}(\Sigma_F)$ this inequality is expressed as

$$\mathrm{P}\left[X \geq \|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)\right] \leq \xi. \tag{G.13}$$

This probability can be also bounded following (G.12) as

$$\mathrm{P}\left[X \geq \|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)\right] \leq \exp(-x^*_{(\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F))}) \leq \xi, \tag{G.14}$$

where $x^*_{(\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F))}$ is the specific form for our problem of (G.11) which is defined as

$$x^*_{(\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F))} = \left(\sqrt{\frac{\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)}{2\lambda_1} + \frac{\sum_{l=1}^{p}\lambda_l^2}{4\lambda_1^2}} - \frac{\sqrt{\sum_{l=1}^{p}\lambda_l^2}}{2\lambda_1}\right)^2, \tag{G.15}$$

with λ_l representing the eigenvalues of the covariance matrix Σ_F and λ_1 representing the biggest of those eigenvalues. Combining (G.14) and (G.15) we find a bound on the trace of Σ_F given by

$$-\left(\sqrt{\frac{\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)}{2\lambda_1} + \frac{\sum_{l=1}^{p}\lambda_l^2}{4\lambda_1^2}} - \frac{\sqrt{\sum_{l=1}^{p}\lambda_l^2}}{2\lambda_1}\right)^2 \leq \ln(\xi)$$

$$\sqrt{\frac{\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)}{2\lambda_1} + \frac{\sum_{l=1}^p \lambda_l^2}{4\lambda_1^2}} - \frac{\sqrt{\sum_{l=1}^p \lambda_l^2}}{2\lambda_1} \geq \sqrt{\ln(1/\xi)}$$

$$\frac{\|\mu\|_2^2\delta^2 - \mathrm{tr}(\Sigma_F)}{2\lambda_1} + \frac{\sum_{l=1}^p \lambda_l^2}{4\lambda_1^2} \geq \left(\sqrt{\ln(1/\xi)} + \frac{\sqrt{\sum_{l=1}^p \lambda_l^2}}{2\lambda_1}\right)^2$$

$$\mathrm{tr}(\Sigma_F) \leq \|\mu\|_2^2\delta^2 - 2\left(\lambda_1 \ln(1/\xi) + \sqrt{\ln(1/\xi)}\sqrt{\sum_{l=1}^p \lambda_l^2}\right)$$

Appendix H. Proof of Proposition 5 in Chapter 4

Combining the definition of $z_i^k = x_i^k + \bar{y}^k - \bar{x}^k - u^k$ and the expression for x_i^{k+1} defined in (4.5), we can express the update of $\bar{y}$ in (4.8) as

$$\bar{y}^{k+1} = (1/n)\underset{\hat{y}\in\mathbb{R}^p}{\arg\min}\left\{h(\hat{y}) + (\rho/2n)\|\hat{y} - n(\bar{x}^{k+1} + u^k)\|^2\right\},$$

where $\hat{y} = n\bar{y}$. Then, we can express $\bar{y}^{k+1}$ in terms of its proximal operator $\bar{y}^{k+1} = (1/n)\,\mathbf{prox}_{(n/\rho)h}[n(\bar{x}^{k+1} + u^k)]$, which can be expressed in terms of the gradient of the Moreau Envelope of h, as in (4.5), leading to

$$\bar{y}^{k+1} = (\bar{x}^{k+1} + u^k) - (1/\rho)\nabla h^{n/\rho}\left(n(\bar{x}^{k+1} + u^k)\right). \tag{H.1}$$

Now, expressing the u-update presented in (4.2) in terms of (H.1) gives

$$u^{k+1} = (1/\rho)\nabla h^{n/\rho}\left(n(\bar{x}^{k+1} + u^k)\right). \tag{H.2}$$

Next, we can express (H.1) in terms of z_i^k as

$$\bar{y}^{k+1} = (1/n)\sum_{i=1}^{n}[z_i^k - (1/\rho)\nabla f_i^{1/\rho}(z_i^k)] + u^k - (1/\rho)\nabla h^{n/\rho}\left(n(\bar{x}^{k+1} + u^k)\right),$$

and by inserting the definition of z_i^k we get

$$\bar{y}^{k+1} = \bar{y}^k - 1/(\rho n)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k) - (1/\rho)\nabla h^{n/\rho}(n(\bar{x}^{k+1} + u^k)). \tag{H.3}$$

Taking the average of the definition of z_i^k we get $\bar{z}^k = \bar{y}^k - u^k$, and by inserting it into the average of the x_i-updates given by $\bar{x}^k = \bar{z}^k - 1/(\rho n)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k)$ we get the equality

$$\bar{y}^k - 1/(\rho n)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k) = \bar{x}^{k+1} + u^k. \tag{H.4}$$

Thus, combining (H.3) and (H.4), we obtain

$$\bar{y}^{k+1} = \bar{y}^k - 1/(\rho n)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k) - (1/\rho)\nabla h^{n/\rho}\left(n\bar{y}^k - (1/\rho)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k)\right), \tag{H.5}$$

and the u-update combining (H.2) with (H.4) is expressed as

$$u^{k+1} = (1/\rho)\nabla h^{n/\rho}\left(n\bar{y}^k - (1/\rho)\sum_{i=1}^{n}\nabla f_i^{1/\rho}(z_i^k)\right). \tag{H.6}$$

As presented in Section 4.1, each agent's $\nabla f_i^{1/\rho}(z_i^k)$ is predicted by the GP and this prediction is used by the ADMM algorithm when the coordinator skips a communication round with an agent. This dynamic is expressed in (4.7) with the variable β_i^k, where depending on the communication decision, β_i^k takes the value of $\nabla f_i^{1/\rho}(z_i^k)$ or its predicted value. In the context of our problem, we replace $\nabla f_i^{1/\rho}(z_i^k)$ from the expressions in (H.5) and (H.6) with the dynamics defined in (4.7), giving the ADMM expression

$$\begin{aligned}
x_i^{k+1} &= z_i^k - (1/\rho)\beta_i^k \\
u^{k+1} &= (1/\rho)\nabla h^{n/\rho}\left(n\bar{y}^k - (1/\rho)\sum_{i=1}^{n}\beta_i^k\right) \\
\bar{y}^{k+1} &= \bar{y}^k - 1/(\rho n)\sum_{i=1}^{n}\beta_i^k - u^{k+1}.
\end{aligned}$$

Defining the variable $v^k = n\bar{y}^k - (1/\rho)\sum_{i=1}^{n}\beta_i^k$, we get that the u-update is given by

$$u^{k+1} = (1/\rho)\nabla h^{n/\rho}\left(v^k\right).$$

Appendix I. Proof of Publication for Previously Published Material

Franklin Open 6 (2024) 100080

Contents lists available at ScienceDirect

Franklin Open

journal homepage: www.elsevier.com/locate/fraope

Optimal querying for communication-efficient ADMM using Gaussian process regression ☆

Aldo Duarte [a,*], Truong X. Nghiem [b], Shuangqing Wei [a]

[a] Division of Electrical and Computer Engineering, School of Electrical Engineering and Commputer Science, Louisiana State University, Baton Rouge, LA 70803, United States

[b] School of Informatics, Computing, and Cyber Systems, Northern Arizona University, Flagstaff, AZ 86011, United States

ARTICLE INFO

Keywords:
Gaussian process
ADMM
Distributed optimization
Proximal operator
Communication reduction

ABSTRACT

In distributed optimization schemes consisting of a group of agents connected to a central coordinator, the optimization algorithm often involves the agents solving private local sub-problems and exchanging data frequently with the coordinator to solve the global distributed problem. In those cases, the query-response mechanism usually causes excessive communication costs to the system, necessitating communication reduction in scenarios where communication is costly. Integrating Gaussian processes (GP) as a learning component to the Alternating Direction Method of Multipliers (ADMM) has proven effective in learning each agent's local proximal operator to reduce the required communication exchange. A key element for integrating GP into the ADMM algorithm is the querying mechanism upon which the coordinator decides when communication with an agent is required. In this paper, we formulate a general querying decision framework as an optimization problem that balances reducing the communication cost and decreasing the prediction error. Under this framework, we propose a joint query strategy that takes into account the joint statistics of the query and ADMM variables and the total communication cost of all agents in the presence of uncertainty caused by the GP regression. In addition, we derive three different decision mechanisms that simplify the general framework by making the communication decision for each agent individually. We integrate multiple measures to quantify the trade-off between the communication cost reduction and the optimization solution's accuracy/optimality. The proposed methods can achieve significant communication reduction and good optimization solution accuracy for distributed optimization, as demonstrated by extensive simulations of a distributed sharing problem.

1. Introduction

In a distributed optimization scheme that consists of a group of agents connected to a central coordinator, the optimization algorithm often involves the agents solving private local sub-problems and exchanging data frequently with the coordinator. In many of those schemes, the underlying local sub-problems in the form of *proximal minimization problems* [1] are solved by the agents in response to queries sent by the coordinator. Proximal minimization is suitable for networks with privacy constraints because it prevents each agent's local objective and constraints from being disclosed to the coordinator or other agents. Once the coordinator receives the local proximal minimization solutions from the agents, it uses them to calculate new queries for the agents that keep on driving the agents' solutions to the global solution. Such distributed optimization schemes have been applied to power management for smart buildings and distribution power systems, among other applications, as shown in [2].

The Alternating Direction Method of Multipliers (ADMM) [3] is an algorithm well suited for distributed optimization settings. It has found great success in distributed optimization due to its simplicity of implementation and its suitability for parallelization. As a result, ADMM has found many applications in machine learning problems [4] and other distributed optimization problems [5–8].

The query-response mechanism inherent to distributed optimization algorithms (ADMM included) often requires many iterations before the algorithm converges to a solution. An extensive amount of communication between the coordinator and the agents could make the system unviable in cases where communication is expensive, such as underwater communication for robot formation control [9]. For that reason, reducing communication expenditure is highly desirable, even critical, for the viability of these distributed optimization schemes in real-life applications.

Communication reduction in distributed optimization settings has previously been studied. The authors of [10] presented a hierarchical distributed optimization algorithm for the predictive control of a

☆ This material is based upon work supported by the U.S. National Science Foundation (NSF) under Grant No. 2238296.
* Corresponding author.
E-mail addresses: aduart3@lsu.edu (A. Duarte), Truong.Nghiem@nau.edu (T.X. Nghiem), swei@lsu.edu (S. Wei).

https://doi.org/10.1016/j.fraope.2024.100080
Received 28 August 2023; Received in revised form 2 February 2024; Accepted 25 February 2024
Available online 1 March 2024

Bibliography

[1] T. X. Nghiem, A. Duarte, and S. Wei, "Learning-based Adaptive Quantization for Communication-efficient Distributed Optimization with ADMM," in *Annual Asilomar Conference on Signals, Systems, and Computers*, California, USA, Nov. 2020.

[2] N. Parikh and S. Boyd, "Proximal algorithms," *Foundations and Trends® in Optimization*, vol. 1, no. 3, pp. 127–239, 2014.

[3] T. Yang, X. Yi, J. Wu, Y. Yuan, D. Wu, Z. Meng, Y. Hong, H. Wang, Z. Lin, and K. H. Johansson, "A survey of distributed optimization," *Annual Reviews in Control*, vol. 47, pp. 278–305, 2019.

[4] D. Varagnolo and et al, "Newton-raphson consensus for distributed convex optimization," *IEEE Transactions on Automatic Control*, vol. 61, no. 4, 2016.

[5] A. Gourtani, T.-D. Nguyen, and H. Xu, "A distributionally robust optimization approach for two-stage facility location problems," *EURO Journal on Computational Optimization*, vol. 8, no. 2, 2020.

[6] P. Dvurechensky and et al, "Hyperfast second-order local solvers for efficient statistically preconditioned distributed optimization," *EURO Journal on Computational Optimization*, vol. 10, 2022.

[7] D. Gabay and B. Mercier, "A dual algorithm for the solution of nonlinear variational problems via finite element approximation," *Computers & Mathematics with Applications*, vol. 2, no. 1, pp. 17–40, 1976.

[8] 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.*, 2011.

[9] S. Kumar, R. Jain, and K. Rajawat, "Asynchronous optimization over heterogeneous networks via consensus admm," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 3, no. 1, pp. 114–129, 2017.

[10] X. Cao and K. J. R. Liu, "Dynamic sharing through the admm," *IEEE Transactions on Automatic Control*, vol. 65, no. 5, pp. 2215–2222, 2020.

[11] Z. Liu, P. Dai, H. Xing, Z. Yu, and W. Zhang, "A distributed algorithm for task offloading in vehicular networks with hybrid fog/cloud computing," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–14, 2021.

[12] T. Song, D. Li, Q. Jin, and K. Hirasawa, "Sparse proximal reinforcement learning via nested optimization," *IEEE Transactions on Systems, Man, and Cybernetics:*

Systems, vol. 50, no. 11, pp. 4020–4032, 2020.

[13] V. Yfantis and et al., "Hierarchical distributed optimization of constraint-coupled convex and mixed-integer programs using approximations of the dual function," *EURO Journal on Computational Optimization*, vol. 11, 2023.

[14] R. Zhao, M. Miao, J. Lu, Y. Wang, and D. Li, "Formation control of multiple underwater robots based on ADMM distributed model predictive control," *Ocean Engineering*, vol. 257, p. 111585, 8 2022.

[15] P. Braun, L. Grüne, C. M. Kellett, S. R. Weller, and K. Worthmann, "A distributed optimization algorithm for the predictive control of smart grids," *IEEE Transactions on Automatic Control*, vol. 61, no. 12, pp. 3898–3911, 2016.

[16] V. Smith, S. Forte, C. Ma, M. Takác, M. I. Jordan, and M. Jaggi, "Cocoa: A general framework for communication-efficient distributed optimization," *arXiv preprint arXiv:1611.02189*, 2016.

[17] C. Ma, J. Konecný, M. Jaggi, V. Smith, M. I. Jordan, P. Richtárik, and M. Takác, "Distributed optimization with arbitrary local solvers," *Optimization Methods Software*, vol. 32, no. 4, pp. 813–848, July 2017.

[18] D. Du, X. Li, W. Li, R. Chen, M. Fei, and L. Wu, "Admm-based distributed state estimation of smart grid under data deception and denial of service attacks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 8, 2019.

[19] S. Zhou and G. Y. Li, "Communication-Efficient ADMM-based Federated Learning," *arXiv e-prints*, p. arXiv:2110.15318, Oct. 2021.

[20] W. Li, Y. Liu, Z. Tian, and Q. Ling, "Communication-censored linearized admm for decentralized consensus optimization," *IEEE Transactions on Signal and Information Processing over Networks*, vol. 6, pp. 18–34, 2020.

[21] G. Stathopoulos and C. N. Jones, "A coordinator-driven communication reduction scheme for distributed optimization using the projected gradient method," in *Proceedings of the 17th IEEE European Control Conference, ECC 2018, Limassol, Cyprus*, 2018.

[22] G. Stathopoulos and C. Jones, "Communication reduction in distributed optimization via estimation of the proximal operator," *arXiv preprint arXiv:1803.07143*, 03 2018.

[23] T. X. Nghiem, G. Stathopoulos, and C. Jones, "Learning Proximal Operators with Gaussian Processes," in *Annual Allerton Conference on Communication, Control, and Computing*, Illinois, USA, Oct. 2018.

[24] C.-X. Shi and G.-H. Yang, "Distributed composite optimization over relay-assisted networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 10, pp. 6587–6598, 2021.

[25] Y. Pu, M. N. Zeilinger, and C. N. Jones, "Quantization Design for Distributed Optimization," *IEEE Transactions on Automatic Control*, vol. 62, no. 5, pp. 2107–2120, May 2017.

[26] T. T. Doan, S. T. Maguluri, and J. Romberg, "Fast Convergence Rates of Distributed Subgradient Methods with Adaptive Quantization," *arXiv:1810.13245 [math]*, Oct. 2018.

[27] P. Groot and P. J. Lucas, "Gaussian process regression with censored data using expectation propagation," 01 2012, pp. 115–122.

[28] G. Bottegal, H. Hjalmarsson, and G. Pillonetto, "A new kernel-based approach to system identification with quantized output data," *Automatica*, vol. 85, pp. 145–152, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0005109817303989

[29] L. V. Nguyen, G. Hu, and C. J. Spanos, "Efficient sensor deployments for spatio-temporal environmental monitoring," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, 2020.

[30] Y. Koyano, Y. Ikeda, Y. Oikawa, and Y. Yamasaki, "Recording and playback system of high speed single-bit direct quantized signal with dither," *Research Publishing, Singapore*, 2016.

[31] A. Ghosh and S. Pamarti, "Dithered quantizers with negligible in-band dither power," *ArXiv*, vol. abs/1202.0936, 2012.

[32] H. Zhu and H. Fujimoto, "Overcoming current quantization effects for precise current control by combining dithering techniques and kalman filter," in *IECON 2012 - 38th Annual Conference on IEEE Industrial Electronics Society*, 2012, pp. 3826–3831.

[33] R. Hadad and U. Erez, "Dithered quantization via orthogonal transformations," *IEEE Transactions on Signal Processing*, vol. 64, no. 22, pp. 5887–5900, 2016.

[34] X. Wang, "On Chebyshev functions and Klee functions," *Journal of Mathematical Analysis and Applications*, vol. 368, no. 1, pp. 293–310, 2010.

[35] D. P. Bertsekas, *Convex Optimization Algorithms*. Athena Scientific, 2015.

[36] C. E. Rasmussen and C. K. Williams, *Gaussian processes for machine learning*. MIT press Cambridge, 2006, vol. 1.

[37] E. Solak, R. Murray-Smith, W. E. Leithead, D. J. Leith, and C. E. Rasmussen, "Derivative observations in gaussian process models of dynamic systems," in *Advances in neural information processing systems*, 2003, pp. 1057–1064.

[38] A. Grami, "Chapter 5 - analog-to-digital conversion," in *Introduction to Digital Communications*, A. Grami, Ed. Boston: Academic Press, 2016, pp. 217 – 264. [Online]. Available: http://www.sciencedirect.com/science/article/pii/B9780124076822000053

[39] A. Sripad and D. Snyder, "A necessary and sufficient condition for quantization errors to be uniform and white," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 25, no. 5, pp. 442–448, October 1977.

[40] J. Rapp, R. M. A. Dawson, and V. K. Goyal, "Estimation from quantized gaussian measurements: When and how to use dither," *IEEE Transactions on Signal Processing*, vol. 67, no. 13, pp. 3424–3438, 2019.

[41] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. of the CACSD Conference*, Taipei, Taiwan, 2004.

[42] J. Vanhatalo, J. Riihimäki, J. Hartikainen, P. Jylänki, V. Tolvanen, and A. Vehtari, "GPstuff: Bayesian modeling with gaussian processes," *Journal of Machine Learning Research*, vol. 14, pp. 1175–1179, 2013.

[43] N. A. NAGENDRA. (2013) Ieee 802.11 mac protocol. [Online]. Available: https://www.mathworks.com/matlabcentral/fileexchange/44110-ieee-802-11-mac-protocol

[44] W. Deng and W. Yin, "On the global and linear convergence of the generalized alternating direction method of multipliers," *J. Sci. Comput.*, vol. 66, no. 3, p. 889–916, mar 2016.

[45] Y. Xie and U. V. Shanbhag, "Si-admm: A stochastic inexact admm framework for resolving structured stochastic convex programs," in *2016 Winter Simulation Conference (WSC)*, 2016, pp. 714–725.

[46] ——, "Si-admm: A stochastic inexact admm framework for resolving structured stochastic convex programs," in *2016 Winter Simulation Conference (WSC)*, 2016, pp. 714–725.

[47] C. Grigo and P.-S. Koutsourelakis, "Bayesian model and dimension reduction for uncertainty propagation: Applications in random media," *SIAM/ASA Journal on Uncertainty Quantification*, vol. 7, no. 1, pp. 292–323, jan 2019.

[48] H. Nagao and M. Srivastava, "Fixed width confidence region for the mean of a multivariate normal distribution," *Journal of Multivariate Analysis*, vol. 81, pp. 259–273, 05 2002.

[49] Y. Liu, W. Xu, G. Wu, Z. Tian, and Q. Ling, "Coca: Communication-censored admm for decentralized consensus optimization," in *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, 2018, pp. 33–37.

[50] Y. Pu, M. N. Zeilinger, and C. N. Jones, "Quantization design for unconstrained distributed optimization," in *2015 American Control Conference (ACC)*, 2015, pp. 1229–1234.

[51] A. Janssen and M. de Jong, "Analysis of contention tree algorithms," *IEEE Transactions on Information Theory*, vol. 46, no. 6, pp. 2163–2172, 2000.

[52] B. Laurent and P. Massart, "Adaptive estimation of a quadratic functional by model selection," *The Annals of Statistics*, vol. 28, no. 5, pp. 1302–1338, 2000. [Online]. Available: http://www.jstor.org/stable/2674095

[53] L. Birgé and P. Massart, "Minimum contrast estimators on sieves: Exponential bounds and rates of convergence," *Bernoulli*, vol. 4, no. 3, pp. 329–375, 1998. [Online]. Available: http://www.jstor.org/stable/3318720

Vita

Aldo Duarte was born and raised in the city of Lima in Peru. He received his bachelor's degree majoring in Telecommunication Engineering from Pontificia Universidad Catolica, Lima, Peru in 2012. Afterwards, he worked for four years in the industry designing indoor RF solutions for the major mobile carriers in Peru. He then joined Lousiana State University (LSU) in the fall of 2016 to pursue a Master's degree in Electrical Engineering in the School of Electrical Engineering and Computer Science of the College of Engineering at LSU in Baton Rouge, LA. A few years later, he started his Ph.D. in Electrical Engineering in the same division at LSU, in the summer of 2018.

Currently, he is a doctoral candidate in the School of Electrical Engineering and Computer Science of the College of Engineering at LSU, and his dissertation research with Dr. Shuangqing Wei was to develop a communication-efficient distributed optimization problem solved using the Alternating Direction Method of Multipliers (ADMM) where communications between agents and coordinator are skipped using Gaussian Processes (GP) regression and those communications are affected by uniform quantization. This study was completed at the end of the Summer Semester of 2024.

Aldo has already completed the requirements for the Doctor of Philosophy degree and plans to graduate in the summer of 2024. Mr. Aldo Duarte plans to stay in the USA for a few more years to gain more work experience during his Optional Practical Training (OPT) period.