Consensus and Sectioning-Based ADMM With Norm-1 Regularization for Imaging With a Compressive Reflector Antenna

Juan Heredia-Juesas, Ali Molaei, Luis Tirado, and José Á. Martínez-Lorenzo, *Senior Member, IEEE*

***Abstract*—This paper presents three distributed techniques to find a sparse solution of the under-determined linear problem $\mathbf{g} = \mathbf{H}\mathbf{u}$ with a norm-1 regularization, based on the Alternating Direction Method of Multipliers (ADMM). Each one of these techniques divide the matrix H into submatrices by rows, columns, or both rows and columns, leading to the so-called *consensus*-based ADMM, *sectioning*-based ADMM, and *consensus and sectioning*-based ADMM, respectively. They are validated for a particular millimeter-wave imaging problem based on the use of a Compressive Reflector Antenna (CRA). The CRA is a hardware designed to increase the sensing capacity of an imaging system and reduce the mutual information among measurements, allowing an effective imaging of sparse targets with the use of Compressive Sensing (CS) techniques. In previous works, the *consensus*-based ADMM has been proved to accelerate the imaging process, and the *sectioning*-based ADMM has shown the ability to potentially reduce the amount of information to be exchanged among the computational nodes, based on the system configuration. In this paper, the mathematical formulation and graphical interpretation of these two techniques, together with the *consensus and sectioning*-based ADMM approach, are presented. The imaging quality, the imaging time, the convergence, the communication efficiency among the computational nodes, and the computational complexity are analyzed and compared. The distributed capabilities of the ADMM-based approaches, together with the high sensing capacity of the CRA, allow the imaging of metallic targets in a 3D domain in quasi-real time with a reduced amount of information exchanged among the nodes.**

***Index Terms*—Compressive Antenna, distributed ADMM, node communications, norm-1 regularization, real-time imaging.**

I. Introduction

SEVERAL numerical techniques have been developed in the past decades for solving problems defined by a linear matrix equation [1]

$$\mathbf{g} = \mathbf{H}\mathbf{u}, \tag{1}$$

where $\mathbf{g} \in \mathbb{C}^m$ is the known data, $\mathbf{H} \in \mathbb{C}^{m\times n}$ is the known forward model, and $\mathbf{u} \in \mathbb{C}^n$ is the unknown vector to be determined. These techniques can be classified in direct and iterative methods. Direct methods are capable of finding an exact solution of the equation (if existing) with a finite number of operations; but they may require an impractical amount of time. Iterative methods theoretically converge asymptotically to a solution with an infinite number of iterations; but an approximate solution, depending on the tolerance defined, can be achieved in a reduced amount of time. In both cases, the inversion of the matrix $\mathbf{H}$ or the matrix $\mathbf{H}^*\mathbf{H}$ is a problem that needs to be addressed too, and also direct and iterative methods have been proposed to this end [1]–[3]. Direct methods not only have to deal with a potentially large computational cost, but also with the structure and condition of the matrix, which may cause inversion errors and singularities. Despite the power enhancement of computational units, which reduce the operation times, the increase of data in recent years leads to a preference for iterative methods. Additionally, the presence of uncertainties or noise in the data is better addressed with iterative methods since they find an approximate solution, that is, a solution within bounded limits. These uncertainties can be modeled by adding a noise vector $\mathbf{w} \in \mathbb{C}^m$ to Eqn. (1) as follows:

$$\mathbf{g} = \mathbf{H}\mathbf{u} + \mathbf{w}. \tag{2}$$

Distributed techniques [1], [4]–[10] allow to assign small pieces of information among several computational nodes for solving smaller problems in a parallel and fast fashion, exchanging the results among the nodes for obtaining a final solution. These distributed techniques may relief the computational load and speed up the convergence, but introduce the problem of communication among those computational nodes, which also has to be addressed [11]–[18].

Regarding the properties of the unknown vector $\mathbf{u}$, of interest in the recent years are those under-determined problems ($m \ll n$) in which the solution sought is sparse; that is $\|\mathbf{u}\|_0 \ll n$, where $\|\cdot\|_0$ represents the number of non-zero elements of the vector. These type of problems are generally solved via the use of Compressive Sensing (CS) techniques by adding a norm-1 regularization, such as Bayesian Compressive Sampling (BCS) [19], K-means Singular Value Decomposition (K-SVD) [20], Fast Iterative Shrinkage-Thresholding Algorithm (FISTA) [21], [22],

Manuscript received January 26, 2021; revised June 27, 2021 and September 26, 2021; accepted October 12, 2021. Date of publication November 2, 2021; date of current version November 18, 2021. This work was supported in part by NSF CAREER Program Award No. 1653671 and in part by the Department of Energy Award No. DE-SC0017614. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. U. S. Kamilov. *(Corresponding author: Juan Heredia-Juesas.)*

The authors are with the Departments of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115 USA, and also with the Departments of Mechanical & Industrial Engineering, Northeastern University, Boston, MA 02115 USA (e-mail: j.herediajuesas@northeastern.edu; jmartinez@coe.neu.edu).

Digital Object Identifier 10.1109/TCI.2021.3124360

Nesterov's Algorithm (NESTA) [23], or the norm-1 regularized form of the Alternating Direction Method of Multipliers (ADMM) [4], [24]–[26]. The use of the norm-1 regularization instead of the norm-0 allows to optimize a convex problem instead of a non-convex one, providing the same solution under certain conditions [27], [28].

This paper presents three iterative and distributive optimization techniques based on ADMM, to find a sparse solution of Eqn. (2), when the norm-1 regularization is applied. These techniques exploit the distributed capabilities of ADMM by dividing $\mathbf{H}$ and solving the problem in several computational nodes, with the goal of reaching real or quasi-real time imaging. This is of special importance in large-scale imaging problems, where a non-distributed method in a single node might not have enough computational capabilities. The division by rows has shown in [5] a speed-up for finding a solution. In [29], [30] it has been proved that the division of the matrix by columns highly reduce the amount of information exchanged among those computational units. This paper, as an extension of [31], recalls in detail these two techniques and shows the mathematical formulation and graphical interpretation of the combination of them, introducing more degrees of freedom for designing an appropriate optimization architecture.

The performance of these techniques is validated in a millimeter-wave imaging application through the use of a Compressive Reflector Antenna (CRA). A CRA is a hardware designed for increasing the sensing capacity of imaging systems, allowing a reduced number of measurement collection for performing imaging with the use of norm-1 regularized CS techniques [32]–[34]. In this case $\mathbf{H} \in \mathbb{C}^{N_m \times N_p}$ is called the sensing matrix, $\mathbf{g} \in \mathbb{C}^{N_m}$ is the vector of measurements, and $\mathbf{u} \in \mathbb{C}^{N_p}$ is the unknown vector of reflectivity, where N_m represents the number of measurements collected and N_p the number of pixels in the imaging domain.

This paper is organized as follows: Section II introduces the algorithm, properties, and conditions of ADMM. Section III develops the mathematical formulation, the graphical interpretation, and the convergence process of the three presented methods for solving Eqn. (2):

- *Consensus-Based ADMM:* Dividing the sensing matrix in submatrices by rows.
- *Sectioning-Based ADMM:* Dividing the sensing matrix in submatrices by columns.
- *Consensus and Sectioning-Based ADMM:* Dividing the sensing matrix in submatrices by rows and columns.

Section IV studies the communications among the computational nodes, comparing the amount of information exchanged by one single node at one iteration for the three different techniques. Section V analyzes the computational complexity of the algorithms, as well as the integration of the communications overhead per iteration on the global cost, providing an approach for the selection of the division parameters. Section VI briefly introduces the description and operation of a CRA. The particular configuration and numerical results are shown in Section VII, where the imaging quality, imaging time, convergence, and communication efficiency among the computational nodes are compared and discussed for the three proposed techniques. The paper concludes in Section VIII.

II. General Formulation of ADMM

A. Variable Splitting

Variable splitting is a technique applicable to unconstrained optimization problems whose objective function can be separated as a sum of two functions. It consists of introducing a new variable, making the unconstrained problem over $\mathbf{u} \in \mathbb{C}^n$

$$\underset{\mathbf{u}}{\text{minimize}}\ f_1(\mathbf{u}) + f_2(g(\mathbf{u})) \tag{3}$$

to be re-written as a constrained problem, as follows:

$$\begin{aligned} &\underset{\mathbf{u},\mathbf{v}}{\text{minimize}}\ f_1(\mathbf{u}) + f_2(\mathbf{v}) \\ &\text{s.t.} \qquad g(\mathbf{u}) = \mathbf{v}, \end{aligned} \tag{4}$$

where $g : \mathbb{C}^n \to \mathbb{C}^q$ is the function that constrains the relationship between the original and the new variable $\mathbf{v} \in \mathbb{C}^q$, [35]. Both problems (3) and (4) are equivalent, but the variable splitting method allows to solve (4) in an easier manner than its counterpart (3).

B. Augmented Lagrangian and Method of Multipliers

Consider the equality-constrained convex optimization problem

$$\begin{aligned} &\underset{\mathbf{u}}{\text{minimize}}\ f(\mathbf{u}) \\ &\text{s.t.} \qquad \mathbf{P}\mathbf{u} = \mathbf{c}, \end{aligned} \tag{5}$$

where $\mathbf{P} \in \mathbb{C}^{p \times n}$ and $\mathbf{c} \in \mathbb{C}^p$. The *augmented Lagrangian* of this problem is

$$L_\rho(\mathbf{u}, \mathbf{d}) = f(\mathbf{u}) + \mathbf{d}^T(\mathbf{P}\mathbf{u} - \mathbf{c}) + \frac{\rho}{2}\|\mathbf{P}\mathbf{u} - \mathbf{c}\|_2^2, \tag{6}$$

where $\mathbf{d} \in \mathbb{C}^p$ is the *Lagrangian multiplier* or *dual variable*, and $\rho > 0$ is the *augmented* or *penalty* parameter. Notice that L_0 is the standard Lagrangian of the problem. The advantage of considering this penalty term is that the Lagrangian function can be shown to be differentiable under mild conditions on the objective function, [4]. The *Method of Multipliers* solves the problem (5) by minimizing over $\mathbf{u}$ and evaluating on the equality constraint residual as follows:

$$\mathbf{u}^{(k+1)} := \underset{\mathbf{u}}{\text{argmin}}\ L_\rho\left(\mathbf{u}, \mathbf{d}^{(k)}\right), \tag{7a}$$

$$\mathbf{d}^{(k+1)} := \mathbf{d}^{(k)} + \rho\left(\mathbf{P}\mathbf{u}^{(k+1)} - \mathbf{c}\right). \tag{7b}$$

C. ADMM Formulation

ADMM is an optimization algorithm for convex functions that takes advantage of both the variable decomposition, splitting the objective function into simpler objectives, and the convergence properties of the method of multipliers, which relaxes the conditions of the objective function [4], [24]. The general representation of ADMM takes the following optimization form:

$$\begin{aligned} &\underset{\mathbf{u},\mathbf{v}}{\text{minimize}}\ f_1(\mathbf{u}) + f_2(\mathbf{v}) \\ &\text{s.t.} \qquad \mathbf{P}\mathbf{u} + \mathbf{Q}\mathbf{v} = \mathbf{c}, \end{aligned} \tag{8}$$

where the known matrices $\mathbf{P} \in \mathbb{C}^{p \times n}$ and $\mathbf{Q} \in \mathbb{C}^{p \times q}$, and vector $\mathbf{c} \in \mathbb{C}^{p}$ determine the constraint over the unknown variable vectors $\mathbf{u} \in \mathbb{C}^{n}$ and $\mathbf{v} \in \mathbb{C}^{q}$. The convex functions f_1 and f_2 have to be extended real valued functions, that is

$$f_1 : \mathbb{C}^{n} \to \mathbb{R} \cup \{+\infty\}, \tag{9a}$$

$$f_2 : \mathbb{C}^{q} \to \mathbb{R} \cup \{+\infty\}, \tag{9b}$$

and they have to be closed and proper, namely, their effective domain (non-infinity values) has to be non-empty and they never reach $-\infty$, mathematically:

$$\exists \mathbf{u} \in Dom\{f\} \mid f(\mathbf{u}) < +\infty, \text{ and} \tag{10a}$$

$$f(\mathbf{u}) > -\infty, \; \forall \mathbf{u} \in Dom\{f\}, \tag{10b}$$

The optimal value of (8) may be denoted by $\mathbf{t}^{\star}$ as

$$\mathbf{t}^{\star} = \inf\{f_1(\mathbf{u}) + f_2(\mathbf{v}) \mid \mathbf{Pu} + \mathbf{Qv} = \mathbf{c}\}. \tag{11}$$

Taking advantage of the method of multipliers [36], the augmented Lagrangian form of this problem is defined as follows:

$$L_{\rho}(\mathbf{u}, \mathbf{v}, \mathbf{d}) = f_1(\mathbf{u}) + f_2(\mathbf{v}) + \\ + \mathbf{d}^{T}(\mathbf{Pu} + \mathbf{Qv} - \mathbf{c}) + \frac{\rho}{2}\|\mathbf{Pu} + \mathbf{Qv} - \mathbf{c}\|_2^2, \tag{12}$$

where $\mathbf{d} \in \mathbb{C}^{p}$ is the dual variable, and ρ is the augmented parameter. A more convenient expression of the augmented Lagrangian can be achieved by the following simple algebraic transformation:

$$\mathbf{d}^{T}\mathbf{r} + \frac{\rho}{2}\|\mathbf{r}\|_2^2 = \frac{\rho}{2}\|\mathbf{r} + \mathbf{s}\|_2^2 - \frac{\rho}{2}\|\mathbf{s}\|_2^2, \tag{13}$$

for $\mathbf{r} = \mathbf{Pu} + \mathbf{Qv} - \mathbf{c}$, and $\mathbf{s} = \frac{1}{\rho}\mathbf{d}$ being the scaled dual variable. Based on this, the general iterative algorithm of ADMM is described as

$$\mathbf{u}^{(k+1)} := \underset{\mathbf{u}}{\operatorname{argmin}} \, L_{\rho}\left(\mathbf{u}, \mathbf{v}^{(k)}, \mathbf{s}^{(k)}\right), \tag{14a}$$

$$\mathbf{v}^{(k+1)} := \underset{\mathbf{v}}{\operatorname{argmin}} \, L_{\rho}\left(\mathbf{u}^{(k+1)}, \mathbf{v}, \mathbf{s}^{(k)}\right), \tag{14b}$$

$$\mathbf{s}^{(k+1)} := \mathbf{s}^{(k)} + \left(\mathbf{Pu}^{(k+1)} + \mathbf{Qv}^{(k+1)} - \mathbf{c}\right). \tag{14c}$$

The fact that f_1 and f_2 are defined over different variables allows the optimization of $\mathbf{u}$ and $\mathbf{v}$ in an *alternating direction* fashion.

Two metrics are defined for evaluating the convergence of the ADMM algorithm. The *primal residual*, which measures the residual of the constraint; and the *dual residual*, which measures the residual of the dual variable optimization between two consecutive iterations; are defined, respectively at iteration k, as follows [4]:

$$\mathbf{r}_{\mathbf{p}}^{(k)} = \mathbf{Pu}^{(k)} + \mathbf{Qv}^{(k)} - \mathbf{c}, \tag{15a}$$

$$\mathbf{r}_{\mathbf{d}}^{(k)} = \rho \mathbf{P}^{T}\mathbf{Q}\left(\mathbf{v}^{(k)} - \mathbf{v}^{(k-1)}\right) \tag{15b}$$

III. ADMM Distributed Solving Methods

ADMM is a convenient method when applying CS for finding a solution of Eqn. (2). Under the assumption that the sensing matrix $\mathbf{H}$ satisfies the Restricted Isometry Property (RIP) [27], [28], [37], [38], and that the unknown vector $\mathbf{u}$ is sparse—that is, the number of non-zero elements N_{nz} is much smaller than the total number of elements, $N_{nz} \ll n$—a sparse solution of Eqn. (2) can be found by minimizing the sum of the convex function $f_1(\mathbf{u}) = \frac{1}{2}\|\mathbf{Hu} - \mathbf{g}\|_2^2$ and the norm-1 regularization $f_2(\mathbf{v}) = \lambda\|\mathbf{v}\|_1$. The particular ADMM formulation for solving Eqn. (2), takes the *lasso* form:

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\|\mathbf{Hu} - \mathbf{g}\|_2^2 + \lambda\|\mathbf{v}\|_1 \\ \text{s.t.} & \mathbf{u} - \mathbf{v} = \mathbf{0}. \end{array} \tag{16}$$

The constraint—defined with $\mathbf{P} = \mathbf{I}_n$, $\mathbf{Q} = -\mathbf{I}_n$, and $\mathbf{c} = \mathbf{0}$—enforces the variables $\mathbf{u}$ and $\mathbf{v}$ to be equal. This optimization problem assumes $\mathbf{w}$ to be white gaussian noise, but other types of noise could be handled by modifying the first term.

Since the dimensions of the sensing matrix $\mathbf{H}$ could be very large—having many pixels in the imaging domain and/or many collected measurements—, a direct resolution of the problem (16) is not usually efficient. Some techniques have been proposed for solving this problem in a distributed fashion using the ADMM, such as [5], [29], for solving fast imaging problems; or [7], [13], for solving a communications problem in the dual space. In this paper, three different methods, focused on solving imaging problems in the primal space, are presented. The aim is to find a sparse solution of Eqn. (2), while reducing the amount of information exchanged among the nodes, the computational complexity, and the imaging time.

A. Consensus-Based ADMM: Row-Wise Division

As presented in [5], problem (16) can be solved in a distributed fashion, by splitting the original matrix $\mathbf{H}$ into M submatrices $\mathbf{H}_i \in \mathbb{C}^{\frac{N_m}{M} \times N_p}$ in a row division, and the vector of measurements $\mathbf{g}$ into M subvectors $\mathbf{g}_i \in \mathbb{C}^{\frac{N_m}{M}}$, as shown in Fig. 1(a). Then, M different under-determined problems $\mathbf{H}_i\mathbf{u} = \mathbf{g}_i$, for $i = 1, \ldots, M$, need to be solved. In particular, the summation of all of them may be optimized together with the norm-1 regularization as follows (Please, refer to Appx. A for the correct interpretation of the notation throughout the whole mathematical process):

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum\limits_{i=1}^{M}\|\mathbf{H}_i\mathbf{u} - \mathbf{g}_i\|_2^2 + \lambda\|\mathbf{v}\|_1 \\ \text{s.t.} & \mathbf{u} = \mathbf{v}. \end{array} \tag{17}$$

In order to make the optimizations independent, M replicas of the unknown variable $\mathbf{u}$ may be defined as $\mathbf{u}^{i}$ for $i = 1, \ldots, M$, turning the expression (17) into

$$\begin{array}{ll} \text{minimize} & \frac{1}{2}\sum\limits_{i=1}^{M}\left\|\mathbf{H}_i\mathbf{u}^{i} - \mathbf{g}_i\right\|_2^2 + \lambda\|\mathbf{v}\|_1 \\ \text{s.t.} & \mathbf{u}^{i} = \mathbf{v}, \; \forall i \in \{1, \ldots, M\}. \end{array} \tag{18}$$

The augmented Lagrangian function for this problem is as follows:

$$L_{\rho}\left(\mathbf{u}^{1}, \ldots, \mathbf{u}^{M}, \mathbf{v}, \mathbf{s}^{1}, \ldots, \mathbf{s}^{M}\right) = \\ = \frac{1}{2}\sum_{i=1}^{M}\left\|\mathbf{H}_i\mathbf{u}^{i} - \mathbf{g}_i\right\|_2^2 + \lambda\|\mathbf{v}\|_1 +$$

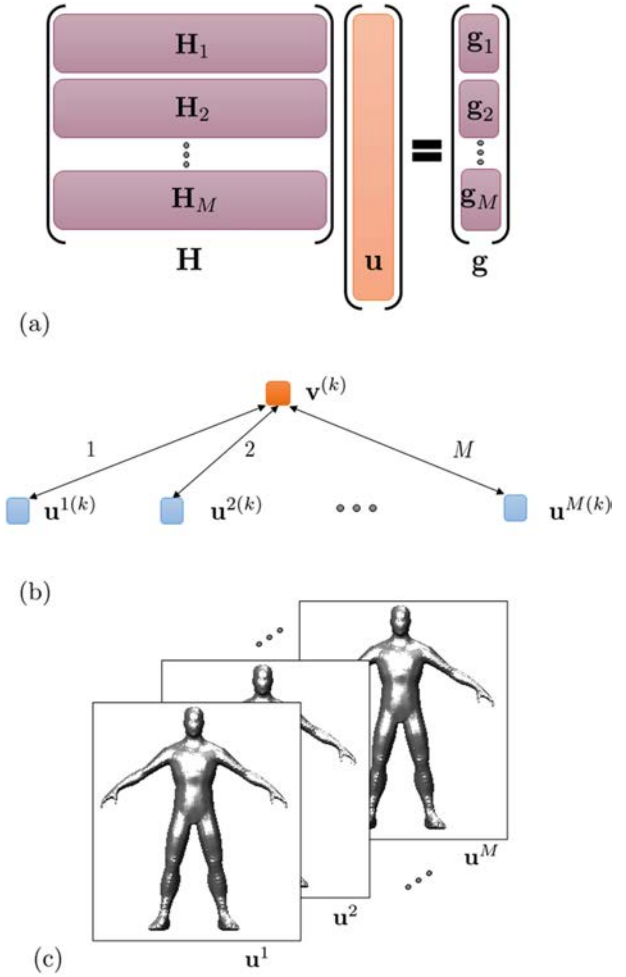


Fig. 1. (a) Division of the matrix equation system by rows. (b) Architecture of the consensus-based ADMM: a central node collects the updates of M sub-nodes, computes the soft-thresholding operator of the mean of them, and then distributes the solution again to the sub-nodes. (c) Graphical interpretation of the row-wise division: M independent images are optimized with few data allocated to each node. The final imaging is a non-linear average of all of them.

$$+\frac{\rho}{2}\sum_{i=1}^{M}\left\|\mathbf{u}^i-\mathbf{v}+\mathbf{s}^i\right\|_2^2-\frac{\rho}{2}\sum_{i=1}^{M}\left\|\mathbf{s}^i\right\|_2^2, \tag{19}$$

where a dual variable $\mathbf{s}^i$ is introduced for each of the M constraints. The augmented parameter ρ enforces the convexity of the Lagrangian function. By iterating the following scheme, an optimal solution may be found:

$$\mathbf{u}^{i,(k+1)}=\left(\mathbf{H}_i^*\mathbf{H}_i+\rho\mathbf{I}_{N_p}\right)^{-1}\left(\mathbf{H}_i^*\mathbf{g}_i+\rho\left(\mathbf{v}^{(k)}-\mathbf{s}^{i,(k)}\right)\right), \tag{20a}$$

$$\mathbf{v}^{(k+1)}=\mathbf{S}_{\frac{\lambda}{M\rho}}\left(\bar{\mathbf{u}}^{(k+1)}+\bar{\mathbf{s}}^{(k)}\right), \tag{20b}$$

$$\mathbf{s}^{i,(k+1)}=\mathbf{s}^{i,(k)}+\mathbf{u}^{i,(k+1)}-\mathbf{v}^{(k+1)}, \tag{20c}$$

where $\bar{\mathbf{u}}$ and $\bar{\mathbf{s}}$ represent the mean of $\mathbf{u}^i$ and $\mathbf{s}^i$, respectively, for all values of i; $\mathbf{I}_{N_p}$ indicate the identity matrix of size N_p; and $\mathbf{S}_\kappa(a)$ is the element-wise soft thresholding operator [39]:

$$\mathbf{S}_\kappa(a)=\begin{cases}a-\kappa\,\mathrm{sign}(a), & |a|>\kappa\\ 0 & |a|\leq\kappa.\end{cases} \tag{21}$$

The *matrix inversion lemma* [40] may be applied for the computation of the term $(\mathbf{H}_i^*\mathbf{H}_i+\rho\mathbf{I}_{N_p})^{-1}$, as shown in Eqn. (22). Therefore, just inverting M matrices of reduced size $\frac{N_m}{M}\times\frac{N_m}{M}$, instead of M large matrices of size $N_p\times N_p$, is required, highly accelerating the algorithm.

$$\left(\mathbf{H}_i^*\mathbf{H}_i+\rho\mathbf{I}_{N_p}\right)^{-1}=\frac{\mathbf{I}_{N_p}}{\rho}-\frac{\mathbf{H}_i^*}{\rho^2}\left(\mathbf{I}_{\frac{N_m}{M}}+\frac{\mathbf{H}_i\mathbf{H}_i^*}{\rho}\right)^{-1}\mathbf{H}_i, \tag{22}$$

In terms of convergence, the primal and dual residuals are computed, respectively, as follows:

$$\mathbf{r}_\mathbf{p}^{(k)}=\left(\mathbf{u}^{1,(k)}-\mathbf{v}^{(k)},\ldots,\mathbf{u}^{M,(k)}-\mathbf{v}^{(k)}\right), \tag{23a}$$

$$\mathbf{r}_\mathbf{d}^{(k)}=-\rho\left(\mathbf{v}^{(k)}-\mathbf{v}^{(k-1)},\ldots,\mathbf{v}^{(k)}-\mathbf{v}^{(k-1)}\right), \tag{23b}$$

and their squared norms are

$$\|\mathbf{r}_\mathbf{p}^{(k)}\|_2^2=\sum_{i=1}^{M}\|\mathbf{u}^{i,(k)}-\mathbf{v}^{(k)}\|_2^2, \tag{24a}$$

$$\|\mathbf{r}_\mathbf{d}^{(k)}\|_2^2=\rho^2M\|\mathbf{v}^{(k)}-\mathbf{v}^{(k-1)}\|_2^2, \tag{24b}$$

It can be noticed in expressions (18) and (20b) that the variable $\mathbf{v}$ acts as a *consensus*, forcing that all variables $\mathbf{u}^i$ converge to the same solution. The primal residual in Eqn. (24a) can be understood, therefore, as a measure of the *lack of consensus*. The architecture of this algorithm can be interpreted as a hierarchical structure, having a central node that collects all individual solution for each sub-node, performs the soft-thresholding averaging, and then broadcasts the global solution to each sub-node, as represented in Fig. 1(b). This technique performs, iteratively, M independent images with few amount of data allocated to each node, and then creates the final imaging as a non-linear average of those intermediate results, in the manner that Fig. 1(c) shows.

As shown in [5], this technique highly reduces the computational cost producing real-time imaging; however, it has the problem of sharing the global solution $\mathbf{v}^{(k)}$ from the central node to each sub-node, and the whole individual solution $\mathbf{u}^{i,(k+1)}$, together with the dual variable $\mathbf{s}^{i,(k)}$, from each sub-node to the central node, for each iteration. These vectors are of the size of the total number of pixels in the imaging domain and may be very large, producing a slow communication among the computational nodes.

B. Sectioning-Based ADMM: Column-Wise Division

A different approach for finding a solution of problem (16) is by dividing the original matrix $\mathbf{H}$ into N submatrices $\mathbf{H}_j\in\mathbb{C}^{N_m\times\frac{N_p}{N}}$ in a column basis and, accordingly, the vector of unknowns $\mathbf{u}$ into N subvectors $\mathbf{u}_j\in\mathbb{C}^{\frac{N_p}{N}}$, as done in [29]. This segmentation makes the problem to be solved in the following form: $\sum_{j=1}^{N}\mathbf{H}_j\mathbf{u}_j=\sum_{j=1}^{N}\hat{\mathbf{g}}_j=\mathbf{g}$, which requires the introduction of the so-called *estimated data* vectors $\hat{\mathbf{g}}_j$, as represented

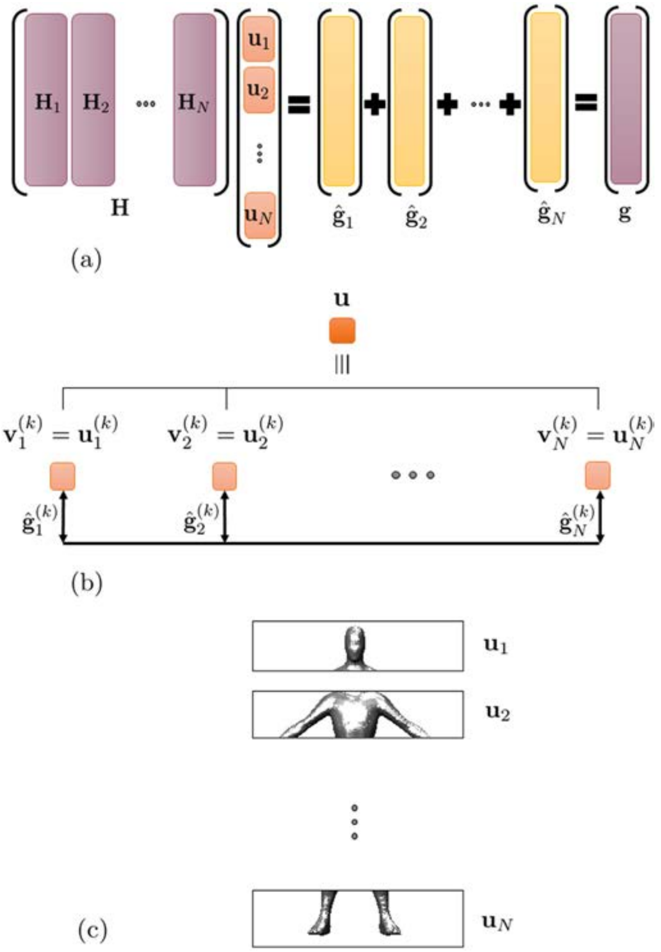


Fig. 2. (a) Division of the matrix equation system by columns. The measurements vector is decomposed into N estimated vectors. (b) Architecture of the sectioning-based ADMM: the problem is split into N nodes that optimize a region of the imaging. For each iteration, they share the small *estimated data* vector with the remaining nodes. (c) Graphical interpretation of the column-wise division: the image is sectioned into N regions. The final imaging is the concatenation of all of them.

in Fig. 2(a). The problem is optimized, together with the norm-1 regularization, as follows:

$$\begin{aligned} \text{minimize } & \tfrac{1}{2}\left\|\sum_{j=1}^{N}\mathbf{H}_j\mathbf{u}_j - \mathbf{g}\right\|_2^2 + \lambda\sum_{j=1}^{N}\|\mathbf{v}_j\|_1 \\ \text{s.t. } & \mathbf{u}_j = \mathbf{v}_j, \quad \forall j \in \{1,\ldots,N\}. \end{aligned} \tag{25}$$

The augmented Lagrangian for this problem is defined over 3 N variables as in the following expression:

$$\begin{aligned} &L_\rho(\mathbf{u}_1,\ldots,\mathbf{u}_N,\mathbf{v}_1,\ldots,\mathbf{v}_N,\mathbf{s}_1,\ldots,\mathbf{s}_N) = \\ &= \frac{1}{2}\left\|\sum_{j=1}^{N}\mathbf{H}_j\mathbf{u}_j - \mathbf{g}\right\|_2^2 + \lambda\sum_{j=1}^{N}\|\mathbf{v}_j\|_1 + \\ &+\frac{\rho}{2}\sum_{j=1}^{N}\|\mathbf{u}_j - \mathbf{v}_j + \mathbf{s}_j\|_2^2 - \frac{\rho}{2}\sum_{j=1}^{N}\|\mathbf{s}_j\|_2^2, \end{aligned} \tag{26}$$

where, again, $\mathbf{s}_j$ is the dual variable introduced for each constraint j, and ρ is the augmented parameter. This problem can be solved by the following iterative scheme:

$$\mathbf{u}_j^{(k+1)} = \left(\mathbf{H}_j^*\mathbf{H}_j + \rho\mathbf{I}_{\frac{N_p}{N}}\right)^{-1}\left(\mathbf{H}_j^*\mathbf{g}_j^{(k)} + \rho\left(\mathbf{v}_j^{(k)} - \mathbf{s}_j^{(k)}\right)\right), \tag{27a}$$

$$\mathbf{v}_j^{(k+1)} = \mathbf{S}_{\frac{\lambda}{\rho}}\left(\mathbf{u}_j^{(k+1)} + \mathbf{s}_j^{(k)}\right), \tag{27b}$$

$$\mathbf{s}_j^{(k+1)} = \mathbf{s}_j^{(k)} + \mathbf{u}_j^{(k+1)} - \mathbf{v}_j^{(k+1)}, \tag{27c}$$

where $\mathbf{g}_j^{(k)}$, required for computing Eqn. (27a), is obtained as

$$\mathbf{g}_j^{(k)} = \mathbf{g} - \sum_{\substack{q=1\\q\neq j}}^{N}\mathbf{H}_q\mathbf{u}_q^{(k)} = \mathbf{g} - \sum_{\substack{q=1\\q\neq j}}^{N}\hat{\mathbf{g}}_q^{(k)}, \tag{28}$$

and it corresponds with the fraction of data determined for the update of the segment j of the vector $\mathbf{u}$, making use of the *estimated data* computed from the remaining segments from the previous iteration. $\mathbf{S}_\kappa(a)$ is the soft thresholding operator as defined in Eqn. (21). In case of $N_m < \frac{N_p}{N}$, the *matrix inversion lemma* can be applied to the term $(\mathbf{H}_j^*\mathbf{H}_j + \rho\mathbf{I}_{\frac{N_p}{N}})^{-1}$ as follows:

$$\left(\mathbf{H}_j^*\mathbf{H}_j + \rho\mathbf{I}_{\frac{N_p}{N}}\right)^{-1} = \frac{\mathbf{I}_{\frac{N_p}{N}}}{\rho} - \frac{\mathbf{H}_j^*}{\rho^2}\left(\mathbf{I}_{N_m} + \frac{\mathbf{H}_j\mathbf{H}_j^*}{\rho}\right)^{-1}\mathbf{H}_j. \tag{29}$$

In this case, only N matrices of sizes $N_m \times N_m$ need to be inverted. However, if $N_m > \frac{N_p}{N}$, the original inversion is computationally more efficient.

In terms of convergence, the primal and dual residual vectors for this technique are computed, respectively, as follows:

$$\mathbf{r}_\mathbf{p}^{(k)} = \left(\mathbf{u}_1^{(k)} - \mathbf{v}_1^{(k)},\ldots,\mathbf{u}_N^{(k)} - \mathbf{v}_N^{(k)}\right), \tag{30a}$$

$$\mathbf{r}_\mathbf{d}^{(k)} = -\rho\left(\mathbf{v}_1^{(k)} - \mathbf{v}_1^{(k-1)},\ldots,\mathbf{v}_N^{(k)} - \mathbf{v}_N^{(k-1)}\right); \tag{30b}$$

and their squared norms are

$$\|\mathbf{r}_\mathbf{p}^{(k)}\|_2^2 = \sum_{j=1}^{N}\|\mathbf{u}_j^{(k)} - \mathbf{v}_j^{(k)}\|_2^2, \tag{31a}$$

$$\|\mathbf{r}_\mathbf{d}^{(k)}\|_2^2 = \rho^2\sum_{j=1}^{N}\|\mathbf{v}_j^{(k)} - \mathbf{v}_j^{(k-1)}\|_2^2. \tag{31b}$$

It is important to notice that the convergence is not guaranteed for this method and depends on ρ, which experimentally should be a large value.

It is deducted from the analysis of Eqns. (27a) and (28) that, for performing the $\mathbf{u}_j^{(k+1)}$ optimizations, each computational node j needs the submatrix $\mathbf{H}_j$, the whole vector $\mathbf{g}$, and the *estimated data* coming from the remaining nodes $\hat{\mathbf{g}}_q^{(k)}$, for $q \neq j$. Therefore, this problem can be interpreted as an N fully-connected net of nodes that individually optimize each fragment $\mathbf{u}_j^{(k+1)}$, introducing thereupon its update to the net in the format of the *estimated data* $\hat{\mathbf{g}}_j^{(k+1)} = \mathbf{H}_j\mathbf{u}_j^{(k+1)} \in \mathbb{C}^{N_m}$, creating a non-hierarchical architecture, as the one represented in Fig. 2(b).

This approach can be illustrated as a *sectioning* of the imaging domain due to splitting the unknown vector $\mathbf{u}$ into N subvectors, corresponding each $\mathbf{u}_j$ to a specific region of the image, as it is schematized in Fig. 2(c). These regions may be predetermined by the user by an appropriate division of the unknown vector $\mathbf{u}$ and, consequently, the matrix $\mathbf{H}$ would be divided accordingly. The final imaging solution is accomplished by connecting, after convergence, the N optimizations $\mathbf{u} = [\mathbf{u}_1; \ldots; \mathbf{u}_N]$.

This technique takes advantage of this image sectioning, since the communication among the nodes requires sharing only small vectors $\hat{\mathbf{g}}_q^{(k)} \in \mathbb{C}^{N_m}$, for each iteration k. However, it lacks the acceleration achieved in the row-wise division due to two main reasons: (i) for small values of N, the inversion of the matrices in Eqn. (29) might be expensive, and (ii) for large values of N, the known vector of measurements $\mathbf{g}$ is highly scattered into the N estimations $\hat{\mathbf{g}}_j$, causing slow computation at each iteration because of the matrix-vector product in Eqns. (27a) and (28).

C. Consensus and Sectioning-Based ADMM: Row and Column-Wise Division

A combination of the two previous approaches may be performed when dividing the matrix $\mathbf{H}$ into $M \cdot N$ submatrices $\mathbf{H}_{ij} \in \mathbb{C}^{\frac{N_m}{M} \times \frac{N_p}{N}}$, the vector of measurements $\mathbf{g}$ into M subvectors $\mathbf{g}_i \in \mathbb{C}^{\frac{N_m}{M}}$, and the unknown vector $\mathbf{u}$ into N subvectors $\mathbf{u}_j \in \mathbb{C}^{\frac{N_p}{N}}$, as shown in Fig. 3. Now, M under-determined problems $\sum_{j=1}^N \mathbf{H}_{ij}\mathbf{u}_j = \sum_{j=1}^N \hat{\mathbf{g}}_{ij} = \mathbf{g}_i$, for $i = 1, \ldots, M$, need to be solved. Applying the same technique as in the division by rows, that is, minimizing the summation of all of them and creating M replicas of each segment j of the unknown vector $\mathbf{u}$, namely $\mathbf{u}_j^i$, the problem may be optimized, together with the norm-1 regularization, as follows:

$$\begin{aligned} \text{minimize} \quad & \frac{1}{2}\sum_{i=1}^{M}\left\|\sum_{j=1}^{N}\mathbf{H}_{ij}\mathbf{u}_j^i - \mathbf{g}_i\right\|_2^2 + \lambda\sum_{j=1}^{N}\|\mathbf{v}_j\|_1 \\ \text{s.t.} \quad & \mathbf{u}_j^i = \mathbf{v}_j, \quad \forall i \in \{1, \ldots, M\}, \quad \forall j \in \{1, \ldots, N\}. \end{aligned} \tag{32}$$

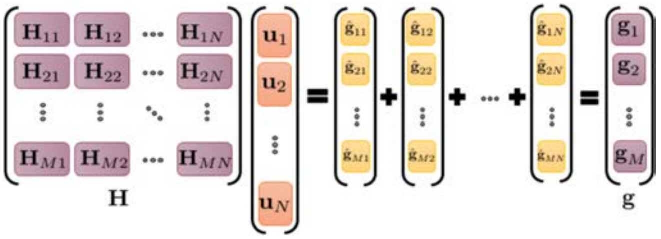
Notice that this problem has $M \cdot N$ equality constraints.

The augmented Lagrangian function for this problem, with $(2M+1)N$ variables, is expressed in the next equation:

$$\begin{aligned} & L_\rho\left(\mathbf{u}_1^1, \ldots, \mathbf{u}_N^M, \mathbf{v}_1, \ldots, \mathbf{v}_N, \mathbf{s}_1^1, \ldots, \mathbf{s}_N^M\right) \\ & = \frac{1}{2}\sum_{i=1}^{M}\left\|\sum_{j=1}^{N}\mathbf{H}_{ij}\mathbf{u}_j^i - \mathbf{g}_i\right\|_2^2 + \lambda\sum_{j=1}^{N}\|\mathbf{v}_j\|_1 \\ & \quad + \frac{\rho}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\|\mathbf{u}_j^i - \mathbf{v}_j + \mathbf{s}_j^i\|_2^2 - \frac{\rho}{2}\sum_{i=1}^{M}\sum_{j=1}^{N}\|\mathbf{s}_j^i\|_2^2, \end{aligned} \tag{33}$$

where $\mathbf{s}_j^i$ is the dual variable for the constraint with indices i and j, and ρ is, as in previous cases, the augmented parameter. This problem can be solved by the following iterative scheme:

$$\mathbf{u}_j^{i,(k+1)} = \left(\mathbf{H}_{ij}^*\mathbf{H}_{ij} + \rho\mathbf{I}_{\frac{N_p}{N}}\right)^{-1}\left(\mathbf{H}_{ij}^*\mathbf{g}_{ij}^{(k)} + \rho\left(\mathbf{v}_j^{(k)} - \mathbf{s}_j^{i,(k)}\right)\right), \tag{34a}$$



$\mathbf{H}_{11}$ $\mathbf{H}_{12}$ ⋯ $\mathbf{H}_{1N}$
$\mathbf{H}_{21}$ $\mathbf{H}_{22}$ ⋯ $\mathbf{H}_{2N}$
$\mathbf{H}_{M1}$ $\mathbf{H}_{M2}$ ⋯ $\mathbf{H}_{MN}$
$\mathbf{H}$
$\mathbf{u}_1$ $\mathbf{u}_2$ $\mathbf{u}_N$
$\hat{\mathbf{g}}_{11}$ $\hat{\mathbf{g}}_{21}$ $\hat{\mathbf{g}}_{M1}$
$\hat{\mathbf{g}}_{12}$ $\hat{\mathbf{g}}_{22}$ $\hat{\mathbf{g}}_{M2}$
$\hat{\mathbf{g}}_{1N}$ $\hat{\mathbf{g}}_{2N}$ $\hat{\mathbf{g}}_{MN}$
$\mathbf{g}_1$ $\mathbf{g}_2$ $\mathbf{g}_M$
$\mathbf{g}$

Fig. 3. Division of the matrix equation system by rows and columns. The measurements vector is divided into M subvectors and each of them is decomposed into N estimated vectors.

$$\mathbf{v}_j^{(k+1)} = \mathbf{S}_{\frac{\lambda}{M\rho}}\left(\bar{\mathbf{u}}_j^{(k+1)} + \bar{\mathbf{s}}_j^{(k)}\right), \tag{34b}$$

$$\mathbf{s}_j^{i,(k+1)} = \mathbf{s}_j^{i,(k)} + \mathbf{u}_j^{i,(k+1)} - \mathbf{v}_j^{(k+1)}, \tag{34c}$$

where

$$\mathbf{g}_{ij}^{(k)} = \mathbf{g}_i - \sum_{\substack{q=1 \\ q \neq j}}^{N}\mathbf{H}_{iq}\mathbf{u}_q^{i,(k)} = \mathbf{g}_i - \sum_{\substack{q=1 \\ q \neq j}}^{N}\hat{\mathbf{g}}_{iq}^{(k)}, \tag{35}$$

corresponds with the fraction of data determined for the update of the $i-th$ replica of the segment j of the vector $\mathbf{u}$, which requires the *estimated data* computed for the remaining segments of the same replica i. $\mathbf{S}_\kappa(a)$ is the soft thresholding operator as defined in Eqn. (21), and $\bar{\mathbf{u}}_j$ and $\bar{\mathbf{s}}_j$ are the mean of $\mathbf{u}_j^i$ and $\mathbf{s}_j^i$, respectively, for all replicas i of a given segment j. If $\frac{N_m}{M} < \frac{N_p}{N}$, the *matrix inversion lemma* should be applied for the inversion of the term $(\mathbf{H}_{ij}^*\mathbf{H}_{ij} + \rho\mathbf{I}_{\frac{N_p}{N}})^{-1}$, as indicated in the Eqn. (36):

$$\left(\mathbf{H}_{ij}^*\mathbf{H}_{ij} + \rho\mathbf{I}_{\frac{N_p}{N}}\right)^{-1} = \frac{\mathbf{I}_{\frac{N_p}{N}}}{\rho} - \frac{\mathbf{H}_{ij}^*}{\rho^2}\left(\mathbf{I}_{\frac{N_m}{M}} + \frac{\mathbf{H}_{ij}\mathbf{H}_{ij}^*}{\rho}\right)^{-1}\mathbf{H}_{ij}. \tag{36}$$

The primal and dual residuals, which are vectors of $M \cdot N$ components that measure the convergence of the algorithm, are computed, respectively, as follows:

$$\begin{aligned} \mathbf{r}_\mathbf{p}^{(k)} = \Big(& \mathbf{u}_1^{1,(k)} - \mathbf{v}_1^{(k)}, \ \ldots, \mathbf{u}_1^{M,(k)} - \mathbf{v}_1^{(k)}, \\ & \vdots \\ & \mathbf{u}_N^{1,(k)} - \mathbf{v}_N^{(k)}, \ \ldots, \mathbf{u}_N^{M,(k)} - \mathbf{v}_N^{(k)}\Big), \end{aligned} \tag{37a}$$

$$\begin{aligned} \mathbf{r}_\mathbf{d}^{(k)} = -\rho\Big(& \mathbf{v}_1^{(k)} - \mathbf{v}_1^{(k-1)}, \ \ldots, \mathbf{v}_1^{(k)} - \mathbf{v}_1^{(k-1)} \\ & \vdots \\ & \mathbf{v}_N^{(k)} - \mathbf{v}_N^{(k-1)}, \ \ldots, \mathbf{v}_N^{(k)} - \mathbf{v}_N^{(k-1)}\Big), \end{aligned} \tag{37b}$$

and their squared norms are

$$\|\mathbf{r}_\mathbf{p}^{(k)}\|_2^2 = \sum_{i=1}^{M}\sum_{j=1}^{N}\|\mathbf{u}_j^{i,(k)} - \mathbf{v}_j^{(k)}\|_2^2, \tag{38a}$$

$$\|\mathbf{r}_\mathbf{d}^{(k)}\|_2^2 = \rho^2 M\sum_{j=1}^{N}\|\mathbf{v}_j^{(k)} - \mathbf{v}_j^{(k-1)}\|_2^2. \tag{38b}$$

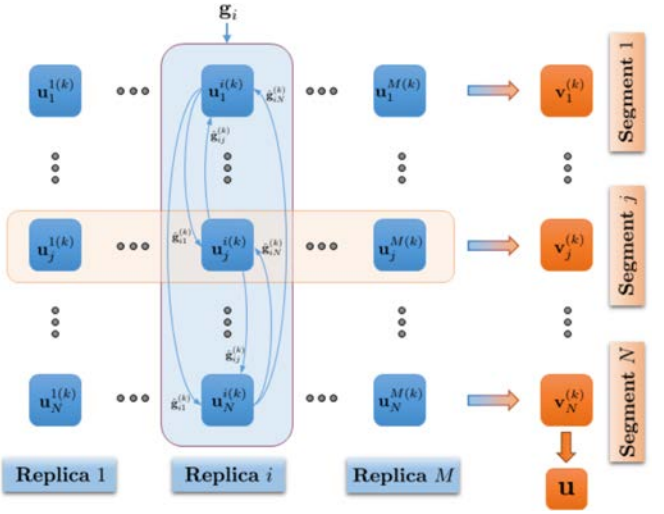


Fig. 4. Schematic of the rows and columns-wise division resolution process. The unknown vector $\mathbf{u}$ is divided into N segments and replicated M times. For a fixed replica i, the optimization of each sub-variable $\mathbf{u}_j^{i,(k)}$ for $j = 1, \ldots, N$ requires the knowledge of the subvector $\mathbf{g}_i$ and the *estimated data* $\hat{\mathbf{g}}_{iq}^{(k)}$ obtained from the previous optimizations of the remaining sub-variables $\mathbf{u}_q^{i,(k)}$ for $q = 1, \ldots, N$ with $q \neq j$. For a given segment j, the sub-variable $\mathbf{v}_j^{(k)}$ acts as the *consensus* of all the replicas $\mathbf{u}_j^{i,(k)}$, for $i = 1 \ldots, M$, of that segment.

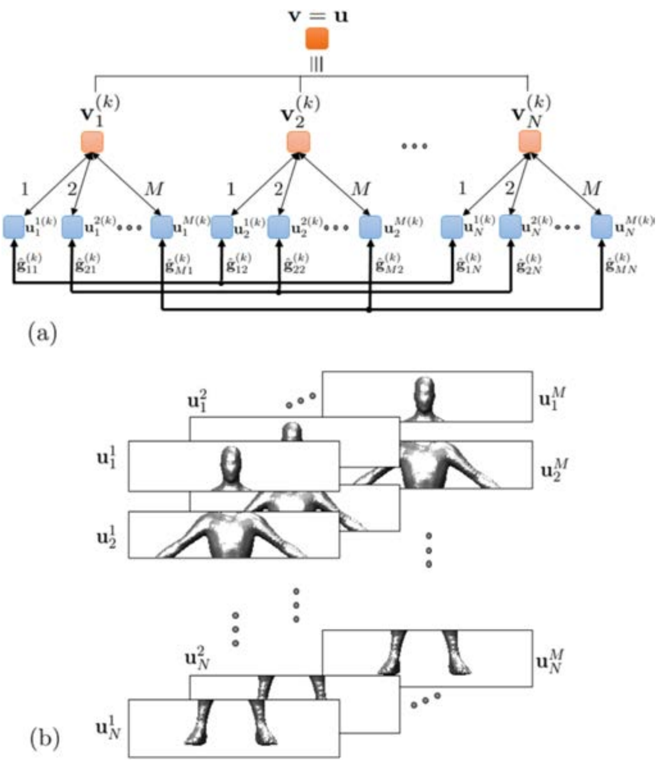


Fig. 5. (a) Architecture of the consensus and sectioning-based ADMM: the problem is split into N nodes, each of them acting as a central node that collects the updates of M sub-nodes, computes the soft-thresholding operator of the mean of them, and then broadcast the solution again to the sub-nodes. Each sub-node shares, for each iteration, the small *estimated data* vector with the remaining sub-nodes that correspond with the same replica. (b) Graphical interpretation of the row and column-wise division: the image is sectioned into N regions, and each of them is replicated M times for performing the imaging with few data allocated to each node. The solution for each region is a non-linear average of all the replicas. The final imaging solution is the concatenation of all the regions.

As in the sectioning-based method, the convergence is not guaranteed and depends on ρ, which experimentally should be a large value.

Equations (32)-(36) combine the particularities of both previous approaches for solving the original problem introduced in Eqn. (2). The matrix $\mathbf{H}$ is divided in submatrices by rows (i indices) and by columns (j indices). For this reason, the unknown vector $\mathbf{u}$ is divided into N segments $[\mathbf{u}_1; \ldots; \mathbf{u}_N]$ and each of them is replicated M times ($\mathbf{u}_j^1, \ldots, \mathbf{u}_j^M$, for $j = 1, \ldots, N$).

For solving this problem, there are two steps in which some information needs to be shared. On one hand, Eqns. (32) and (34b) show that, for a given segment j, $\mathbf{v}_j^{(k+1)}$ acts as a *consensus* variable, imposing the agreement among all $\mathbf{u}_j^{i,(k+1)}$ for $i = 1, \ldots, M$, namely, among all replicas of the same segment. On the other hand, Eqns. (34a) and (35) show that, for a given replica i, the optimization of the variables $\mathbf{u}_j^{i,(k+1)}$ for $j = 1, \ldots, N$, that is, the optimization of all segments of the same replica, require the knowledge of the subvector $\mathbf{g}_i$, as well as the updates of the *estimated data* $\hat{\mathbf{g}}_{iq}^{(k)} = \mathbf{H}_{iq}\mathbf{u}_q^{i,(k)} \in \mathbb{C}^{N_m}$ for $q = 1, \ldots, N$ with $q \neq j$, from the previous iteration. This explanation is depicted in Fig. 4.

Therefore, as Fig. 5(a) shows, this technique can be seen as a net formed by N main nodes. Each of them acts as the central node for optimizing a section of the image, collecting the individual solution of M sub-nodes that perform the imaging of each replica, and computing the soft-thresholding averaging. Then, they broadcast the global result to each sub-node. There are a total of $N \cdot M$ sub-nodes, each one containing a small portion of information $\mathbf{H}_{ij}$ of the general matrix $\mathbf{H}$. For a given replica i, all sub-nodes have to be in communication to exchange their particular *estimated data* $\hat{\mathbf{g}}_{ij}^{(k)} = \mathbf{H}_{ij}\mathbf{u}_j^{i,(k)}$. The final imaging solution is performed by connecting the N different solutions from each central node, $\mathbf{v} = [\mathbf{v}_1; \ldots; \mathbf{v}_N]$.

As graphically shown in Fig. 5(b), this technique *sections* the imaging domain into N small regions. For each of them, M independent images are performed with less data each one. The final imaging for each region is computed as a non linear average of these independent images. Finally, the global imaging solution is the re-connection of all those regions.

In this sense, this technique combines the advantages of both previous techniques: (i) by dividing by rows, the process is accelerated since small optimizations are performed in a parallel fashion; (ii) by dividing by columns, small vectors have to be shared among the nodes of the same replica; and (iii) when combining the division by rows and by columns, the size of the vectors to be exchanged among the computational nodes of the net could be reduced even more, alleviating the communication overhead. A detailed analysis of the communication among the nodes is explained in Section IV. These two degrees of freedom enable performing the optimization in a fast and distributed fashion, making the imaging of large domains feasible.

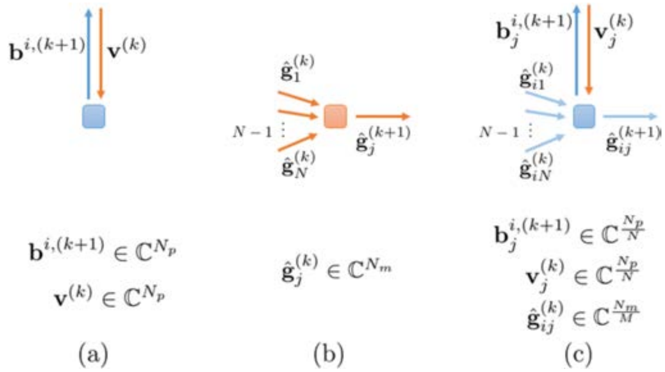

$\mathbf{b}^{i,(k+1)}$
$\mathbf{v}^{(k)}$
$\mathbf{b}^{i,(k+1)} \in \mathbb{C}^{N_p}$
$\mathbf{v}^{(k)} \in \mathbb{C}^{N_p}$
(a)
$\hat{\mathbf{g}}_1^{(k)}$
$N-1$
$\hat{\mathbf{g}}_N^{(k)}$
$\hat{\mathbf{g}}_j^{(k+1)}$
$\hat{\mathbf{g}}_j^{(k)} \in \mathbb{C}^{N_m}$
(b)
$\mathbf{b}_j^{i,(k+1)}$
$\mathbf{v}_j^{(k)}$
$\hat{\mathbf{g}}_{i1}^{(k)}$
$N-1$
$\hat{\mathbf{g}}_{iN}^{(k)}$
$\hat{\mathbf{g}}_{ij}^{(k+1)}$
$\mathbf{b}_j^{i,(k+1)} \in \mathbb{C}^{\frac{N_p}{N}}$
$\mathbf{v}_j^{(k)} \in \mathbb{C}^{\frac{N_p}{N}}$
$\hat{\mathbf{g}}_{ij}^{(k)} \in \mathbb{C}^{\frac{N_m}{M}}$
(c)

Fig. 6. Schematic representation of the vectors and their lengths that are received from and transmitted by one single node at iteration k when the sensing matrix of the problem is divided into submatrices (a) by rows, (b) by columns, and (c) by both rows and columns.

IV. Communication Among the Nodes for the ADMM Solution Techniques

A. Exchange of Information for One Single Node

The three techniques studied in Section III present three distributed ways for finding an optimal solution of expression (16), in which several computational nodes optimize independent sub-problems with few information allocated to each one. However, in all these three methodologies, there are concrete steps in which some information needs to be exchanged. In this section, the amount of data that is transmitted from and received by one single node at iteration k is analyzed for the three techniques:

- In the case of dividing the sensing matrix by rows (Fig. 6(a)), each node i in the lower level has to receive the last version of $\mathbf{v}^{(k)} \in \mathbb{C}^{N_p}$, and then, after the optimization, it has to send its whole new updated version $\mathbf{u}^{i,(k+1)}$, together with the dual variable $\mathbf{s}^{i,(k)}$, to the main node (recall Eqns. in (20) and Fig. 1(b)) in the form $\mathbf{b}^{i,(k+1)} = \mathbf{u}^{i,(k+1)} + \mathbf{s}^{i,(k)} \in \mathbb{C}^{N_p}$. The exchange of information is performed in terms of the imaging and, therefore, a total of $2N_p$ elements need to be exchanged at each iteration.
- In the case of dividing the sensing matrix by columns (Fig. 6(b)), each sub-node j of the lower level receives the *estimated data* of the remaining $N-1$ nodes $\hat{\mathbf{g}}_q^{(k)} \in \mathbb{C}^{N_m}$ for $q = 1, \ldots, N$, with $q \neq j$, and also it broadcasts its own *estimated data* $\hat{\mathbf{g}}_j^{(k+1)} \in \mathbb{C}^{N_m}$ to the remaining nodes (See Eqns. (27a) and (28), and Fig. 2(b)). Since the exchange of information is carried out in terms of the *estimated data*, a total of $N \cdot N_m$ elements are shared by one node at each iteration.
- In the case of performing the division of the sensing matrix in both rows and columns (Fig. 6(c)), as recalled, the unknown vector $\mathbf{u}$ is divided into N segments and each of them is replicated M times. The sub-node ij, which optimizes the replica i of the segment j in the lower level, receives the latest version of $\mathbf{v}_j^{(k)} \in \mathbb{C}^{\frac{N_p}{N}}$ and $N-1$ *estimated data* subvectors $\hat{\mathbf{g}}_{iq}^{(k)} \in \mathbb{C}^{\frac{N_m}{M}}$ for $q = 1, \ldots, N$, with $q \neq j$. Once the variable $\mathbf{u}_j^{i,(k+1)}$ is updated, it sends $\mathbf{b}_j^{i,(k+1)} = \mathbf{u}_j^{i,(k+1)} + \mathbf{s}_j^{i,(k)} \in \mathbb{C}^{\frac{N_p}{N}}$ to the central node of the segment j, and also it broadcasts its own *estimated data* subvector $\hat{\mathbf{g}}_{ij}^{(k+1)} \in \mathbb{C}^{\frac{N_m}{M}}$ to the remaining nodes (recall Eqns. (34), (35), and Fig. 5(a)). Summarizing, at each iteration, a total of $N\frac{N_m}{M} + 2\frac{N_p}{N}$ elements are exchanged by one single node. In this case, the exchange of information is done as a combination of the imaging domain and the *estimated data*.

TABLE I
Number of Elements Exchanged by One Single Node at One Iteration for the Three ADMM Distributed Techniques

ADMM method	**# of elements shared for one node at iteration k**
Consensus-based (Row-wise division)	$2N_p$
Sectioning-based (Column-wise division)	$N \cdot N_m$
Consensus and Sectioning-based (Row and column-wise division)	$N\frac{N_m}{M} + 2\frac{N_p}{N}$

Table I shows the amount of elements to be received by and transmitted from one single node at iteration k for the three analyzed cases.

B. Communication Efficiency of the Three Distributed ADMM Techniques

In order to assess the efficiency of the communications among the nodes for the three different techniques, the amount of information received by and transmitted from one single node at iteration k is compared. Since the number of pixels N_p and the number of measurements N_m are always known, the ratio $R = \frac{N_p}{N_m}$ is considered as the reference for the analysis of the three cases.

1) Column-Wise vs Row-Wise Division: The column-wise division (Sectioning-based ADMM) is more efficient than the row-wise division (Consensus-based ADMM) in terms of communications among the nodes if the following inequality is satisfied:

$$N \cdot N_m < 2N_p. \tag{39}$$

This implies that the number of divisions by columns of the sensing matrix has to satisfy

$$1 < N < 2R. \tag{40}$$

Fig. 7 graphically represents this inequality.

2) Row and Column-Wise vs Row-Wise Division: The row and column-wise division (Consensus and sectioning-based ADMM) is more efficient than the row-wise division (Consensus-based ADMM) in terms of communications among the nodes if the following inequality is satisfied:

$$N\frac{N_m}{M} + 2\frac{N_p}{N} < 2N_p, \tag{41}$$

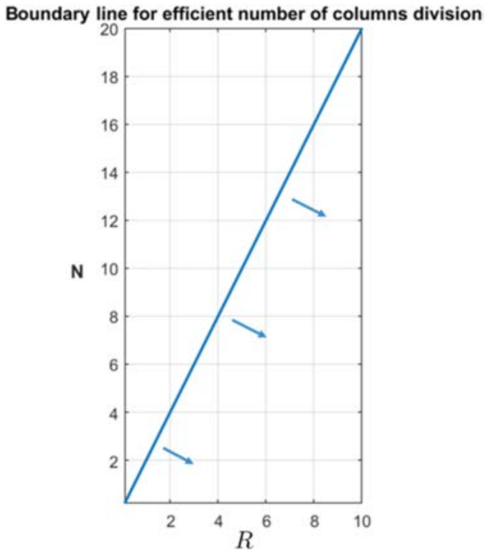


Fig. 7. Boundary line comparing the efficiency of the column-wise division versus the row-wise division. Dividing the sensing matrix into submatrices by columns is more efficient than dividing it by rows, in terms of communications among the nodes, for the integer and positive values of N that fall in the area indicated by the arrows, given $R = \frac{N_p}{N_m}$.

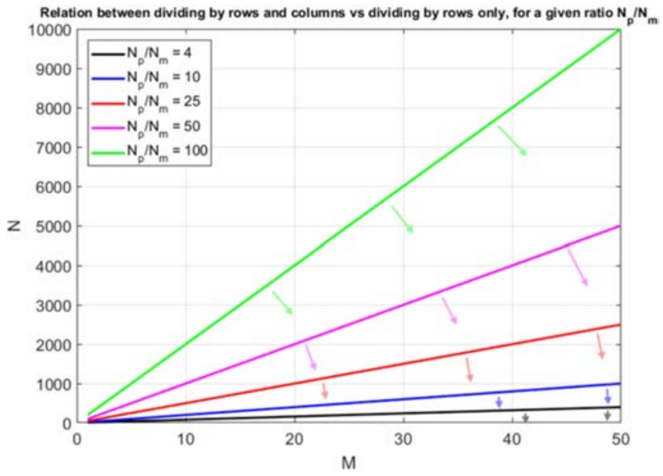


Fig. 8. Boundary curves comparing the efficiency of the row and column-wise division versus the row-wise division. Dividing the sensing matrix in submatrices by rows and columns is more efficient, in terms of communications among the nodes, than dividing it by rows only, for the integer and positive values of M and N that fall in the area indicated by the arrows, for a given ratio $R = \frac{N_p}{N_m}$.

which implies that

$$\frac{N^2}{2M(N-1)} < R. \tag{42}$$

For a given ratio R, the number of column divisions N, in terms of the number of row divisions M, must satisfy the following inequality:

$$1 < N < \sqrt{R^2\,M^2 - 2RM} + RM \sim 2RM. \tag{43}$$

Fig. 8 represents this inequality for some particular ratios R. The division of the sensing matrix by rows and columns is more efficient than the division by rows only, for those integer and positive values of M and N that fall in the region indicated by the arrows. These results illustrate that, in general, for a given ratio R, the division by rows and columns is more efficient the division by rows alone, unless the number of divisions by columns, N, is much larger than the number of divisions by rows, M.

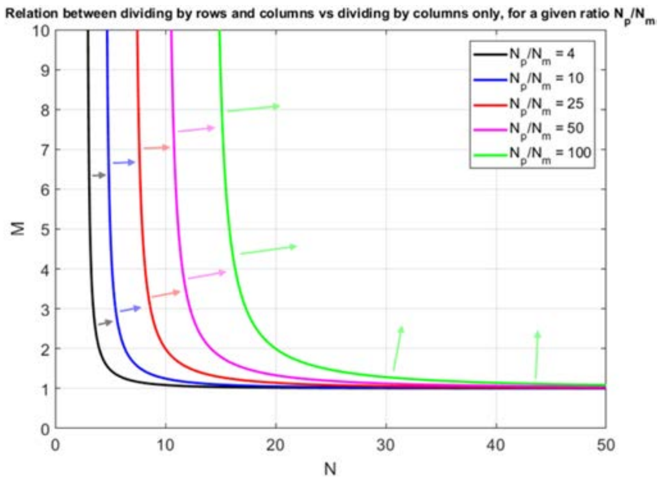


Fig. 9. Boundary curves comparing the efficiency of the row and column-wise division versus the column-wise division. Dividing the sensing matrix into submatrices by rows and columns is more efficient, in terms of communications among the nodes, than dividing it by columns only, for those integer values of N and M that fall in the area indicated by the arrows, for a given ratio $R = \frac{N_p}{N_m}$.

3) Row and Column-Wise vs Column-Wise Division: The row and column-wise division (Consensus and sectioning-based ADMM) is more efficient than the column-wise division (Sectioning-based ADMM) in terms of communications among the nodes if the following inequality is satisfied:

$$N\frac{N_m}{M} + 2\frac{N_p}{N} < N \cdot N_m. \tag{44}$$

This implies that

$$\frac{N^2(M-1)}{2\,M} < R. \tag{45}$$

Therefore, given a ratio R, the number of divisions by rows M in terms of the number of divisions by columns N must satisfy

$$M > \frac{N^2}{N^2 - 2R}. \tag{46}$$

Fig. 9 represents this inequality for some specific ratios R. The division of the sensing matrix by rows and columns is more efficient than the division by columns only for those integer values of N and M that fall in the region indicated by the arrows. Interestingly, the number of divisions by columns, N, has to be large enough so that the division by rows and columns may have a better efficiency than the division by columns only. In fact, given a ratio R, for those values of N such that $N^2 - 2R \leq 0$, the division by columns always will be more efficient than the division by rows and columns.

V. Computational Complexity of the Algorithm

In order to assess and compare the complexity of the three distributed algorithms, an evaluation of the computational cost per iteration is carried out. In this process, the communications overhead described in the previous section is added to this cost to merge, under the same framework, the computational and communication loads. The calculation is based on the optimization scheme of the Consensus and Sectioning-based ADMM from equations (34) to (36), since the Consensus-based and the Sectioning-based algorithms can be particularized from here by setting $N = 1$ and $M = 1$, respectively, although with some particularities.

It can be noticed that Eqns. (34b) and (34c) are just additions and subtractions of vectors, meaning that their computational cost is negligible with respect to Eqn. (34a), which involves matrix-vector multiplications, including the intermediate computation of $\mathbf{g}_{ij}^{(k)}$ in Eqn. (35), which also has matrix-vector multiplication.

Equation (34a) implies a previous computation of the inversion of a matrix that do not depend on the iteration step. The matrix inversion lemma described in Eqn. (36) helps to reduce the computational cost of this inversion but, in order to promote matrix-vector multiplications and reduce matrix-matrix multiplications, Eqn. (34a) is implemented as the sum of four terms, as follows:

$$\begin{aligned}&\left(\frac{\mathbf{I}_{\frac{N_p}{N}}}{\rho} - \frac{\mathbf{H}_{ij}^*}{\rho^2}\left(\mathbf{I}_{\frac{N_m}{M}} + \frac{\mathbf{H}_{ij}\mathbf{H}_{ij}^*}{\rho}\right)^{-1}\mathbf{H}_{ij}\right)\\ &\quad\times\left(\mathbf{H}_{ij}^*\mathbf{g}_{ij}^{(k)} + \rho\left(\mathbf{v}_j^{(k)} - \mathbf{s}_j^{i,(k)}\right)\right)\\ &= \frac{1}{\rho}\mathbf{H}_{ij}^*\mathbf{g}_{ij}^{(k)} + \left(\mathbf{v}_j^{(k)} - \mathbf{s}_j^{i,(k)}\right)\\ &\quad - \frac{1}{\rho^2}\mathbf{F}_{ij}\left(\mathbf{H}_{ij}\left(\mathbf{H}_{ij}^*\mathbf{g}_{ij}^{(k)}\right)\right) - \frac{1}{\rho}\mathbf{F}_{ij}\mathbf{H}_{ij}\left(\mathbf{v}_j^{(k)} - \mathbf{s}_j^{i,(k)}\right),\end{aligned} \tag{47}$$

where

$$\mathbf{F}_{ij} = \mathbf{H}_{ij}^*\left(\mathbf{I}_{\frac{N_m}{M}} + \frac{\mathbf{H}_{ij}\mathbf{H}_{ij}^*}{\rho}\right)^{-1} \in \mathbb{C}^{\frac{N_p}{N}\times\frac{N_m}{M}}. \tag{48}$$

The computation of the $N \cdot M$ matrices $\mathbf{F}_{ij}$ is done once as the first stage of the algorithm, but its computational burden is considerable. It is known that the computational complexity of a matrix-matrix multiplication of sizes $n \times m$ and $m \times p$ is of the order $O(nmp)$, the addition of two vectors of size n is $O(n)$, and the complexity of the inversion of a matrix of size $n \times n$ is of the order $O(n^3)$. Based on this, and neglecting the low order terms, the complexity of the computation of the $N \cdot M$ matrices $\mathbf{F}_{ij}$ is

$$O\left(N\frac{N_m^3}{M^2}\right). \tag{49}$$

On the other hand, the computation of $\mathbf{g}_{ij}^{(k)}$ is $O((N-1)(\frac{N_p}{N}\frac{N_m}{M} + \frac{N_m}{M}) + \frac{N_m}{M})$ and the complexities for each of the four terms in Eqn. (47) are $O(\frac{N_p}{N}\frac{N_m}{M})$, $O(\frac{N_p}{N})$, $O(3\frac{N_p}{N}\frac{N_m}{M})$, and $O(2\frac{N_p}{N}\frac{N_m}{M})$, respectively. All together, plus the sum of the four terms, for the computation of the $N \cdot M$ instances of $\mathbf{u}_j^{i,(k+1)}$ for iteration $k+1$, generates a computational complexity of the order of $O(NN_pN_m + N^2N_m + MN_p)$, when simplifying the constants.

The cost related to the communications among the computational nodes is added to the complexity per iteration, since it is needed to continue the algorithm for the next iteration. Let $\mathcal{C}(P)$ be the cost of exchanging a vector of length P. This cost depends on the particular architecture deployed for the concrete scenario, and may vary depending on the type and speed of communications, but for sure it would be a monotonically increasing function. As it was determined in the previous section, the communications overhead for the Consensus and Sectioning-based ADMM is $N\frac{N_m}{M} + 2\frac{N_p}{N}$, for $N > 1$ and $M > 1$. In this way, the computational cost, together with the communications overhead for one iteration is of the order of

$$O\left(NN_pN_m + N^2N_m + MN_p + \mathcal{C}\left(N\frac{N_m}{M} + 2\frac{N_p}{N}\right)\right). \tag{50}$$

Finally, the computational cost is the same for each iteration, since there is no variation in the optimization scheme that depends on the iteration step. Thus, the complexity grows linearly with the number of iterations.

A. Complexity in Terms of Number of Divisions for the Three Distributed Techniques

Let us consider from this point a fixed system with the freedom of selecting the distributed architecture; to wit, N_p and N_m are given, and M and N are variable. Despite the concrete values of N_p and N_m would influence the complexity, when considering constant, only the variations of M and N are taken into account for the computational part. The analysis could be done as well if there is freedom on the determination of the number of measurements or the degree of discretization of the imaging domain for a constrained distributed architecture. It looks clear that the complexity grows, in general, with the number of divisions, but is better to separate this analysis for the different distributed cases:

- *Consensus-Based ADMM (N = 1):* In this case, the complexity of the matrix inversions reduces with the number of divisions by rows, $O(\frac{1}{M^2})$. In terms of the per-iteration complexity, it is of the order of $O(M + \mathcal{C}(2N_p))$, meaning that it grows linearly with the number of divisions by rows, but is also burdened with a communications cost of a large, constant vector.
- *Sectioning-Based ADMM (M = 1):* This case makes the complexity of the matrix inversion to grow linearly with the number of divisions $O(N)$, and the per-iteration complexity to grow quadratically $O(N^2 + \mathcal{C}(N \cdot N_m))$, weighted by a low cost, but linearly growing, communications term.
- *Consensus and Sectioning-Based ADMM:* The complexity of the matrix inversion for this case is $O(\frac{N}{M^2})$, and the iteration complexity is $O(N^2 + M + \mathcal{C}(N\frac{N_m}{M} + 2\frac{N_p}{N}))$. On one hand, the number of divisions by columns, N,

TABLE II
ALGORITHM COMPLEXITY

ADMM method	Matrix inv. stage	Per-iteration stage
Consensus-based	$O\left(\frac{1}{M^2}\right)$	$O\left(M + \mathcal{C}\left(2N_p\right)\right)$
Sectioning-based	$O(N)$	$O\left(N^2 + \mathcal{C}\left(N \cdot N_m\right)\right)$
Consensus and Sectioning-based	$O\left(\frac{N}{M^2}\right)$	$O\left(N^2 + M + \mathcal{C}\left(\frac{N}{M} N_m + \frac{2N_p}{N}\right)\right)$

increments the complexity but helps to reduce the communications cost, up to a point. On the other hand, the number of divisions by rows, M, helps to reduce the complexity of the matrix inversions and makes the iteration complexity to grow only linearly; but it also plays a trade-off role in the communications cost, reducing the term related to the *data estimation* part, but also introducing a large term related to *consensus* part, although this last one is alleviated by the division by columns, N.

The previous results are summarized in Table II .

B. Complexity Discussion and Selection of Parameters

There is no a standard procedure to establish the optimal selection of the parameters N and M, since, as mentioned previously, it depends on the particular scenario and distributed architecture. However, there are several conclusions that can be taken from the previous analysis. On one hand, it is desired to increase the number of divisions by rows, M to reduce the complexity of the matrix inversions, but not too much since that makes the iteration part to grow. This creates a trade-off between the two stages of the algorithm that should be balanced. On the other hand, the division by columns, N, reduces the communications cost, but increases both the matrix inversion and iteration computation complexities. Therefore, it is required to have the smallest value of N as possible, as long as the communications cost effect is minimized.

VI. Compressive Reflector Antenna

The concept of a Compressive Reflector Antenna (CRA) has been presented recently as a hardware capable of improving the sensing capacity of imaging systems in passive [41], [42] and active [43]–[46] mm-wave radar applications. A way of building a CRA is by distorting the surface of a Traditional Reflector Antenna (TRA) with some scatterers Ω_i, characterized by their dimension $\{D_i^x, D_i^y, D_i^z\}$ and electromagnetic properties: permittivity ϵ_i, permeability μ_i, and conductivity σ_i, as it is shown in Fig. 10, [33]. Other parameters, such as the aperture size D, the focal distance f, and the offset height h_o are in common with a TRA. This distortion modifies the well-known planar phase front pattern of a TRA, creating pseudo-random patterns that can be considered as spatial and spectral codes in the near and far field of the antenna [47]. This phenomenon reduces the mutual information among the measurements, increases the sensing capacity of the system, and allows the use of CS techniques for performing the imaging of sparse 3D objects [32]–[34], [48].

TABLE III
PARAMETERS OF THE NUMERICAL SIMULATION

PARAM.	VALUE	PARAM.	VALUE
f_c	$73.5\ GHz$	Δx_0^T	$30cm$
BW	$7\ GHz$	Δy_0^T	$30cm$
λ_c	$4.1 \cdot 10^{-3} m$	Δz_0^T	$6cm$
D	$50cm$	l_x	λ_c
f	$50cm$	l_y	λ_c
h_o	$35cm$	l_z	$5\lambda_c$
θ_t	$30°$	N_{Tx}	12
$\langle D^x \rangle$	$5\lambda_c$	N_{Rx}	12
$\langle D^y \rangle$	$5\lambda_c$	N_f	15
α_{tmax}	$3°$	N_m	2160
z_0^T	$86cm$	N_p	22500

Based on the configuration depicted in Fig. 11, N_{Tx} transmitting antennas and N_{Rx} receiving antennas are facing the CRA$_1$ and CRA$_2$, respectively. The signal sent from each transmitter is collected by each receiver after being scattered by the targets. The total number of measurements collected is $N_m = N_{Tx} \cdot N_{Rx} \cdot N_f$, where N_f is the total number of equally-spaced frequencies used within a bandwidth of BW around the central frequency f_c. The imaging domain is discretized into N_p pixels. A linear relationship can be established between the vector of measurements $\mathbf{g} \in \mathbb{C}^{N_m}$ and the unknown vector of reflectivity $\mathbf{u} \in \mathbb{C}^{N_p}$ as follows:

$$\mathbf{g} = \mathbf{H}\mathbf{u} + \mathbf{w}, \tag{51}$$

where $\mathbf{H} \in \mathbb{C}^{N_m \times N_p}$ is the sensing matrix computed as described in [49] and $\mathbf{w} \in \mathbb{C}^{N_m}$ is the noise collected for each measurement.

VII. Numerical Results

The effectiveness of the three ADMM techniques is assessed by the use of CRAs for mm-wave imaging applications. Fig. 11(a) shows a schematic of the configuration for the imaging problem. Two h_o-offset CRAs are tilted θ_t and $-\theta_t$ degrees, as shown in Fig. 11(b). The transmitting array is facing CRA$_1$, centered in its focal point and arranged along the $\hat{x}$-axis; meanwhile the receiving array, linearly arranged in the YZ-plane, is centered in the focal point of the CRA$_2$ and facing it. The surfaces of the two CRAs are discretized into triangular patches,

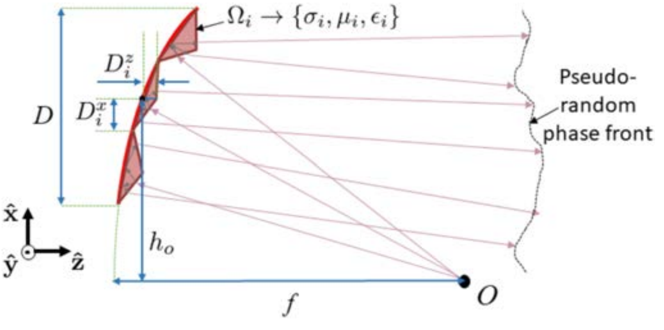
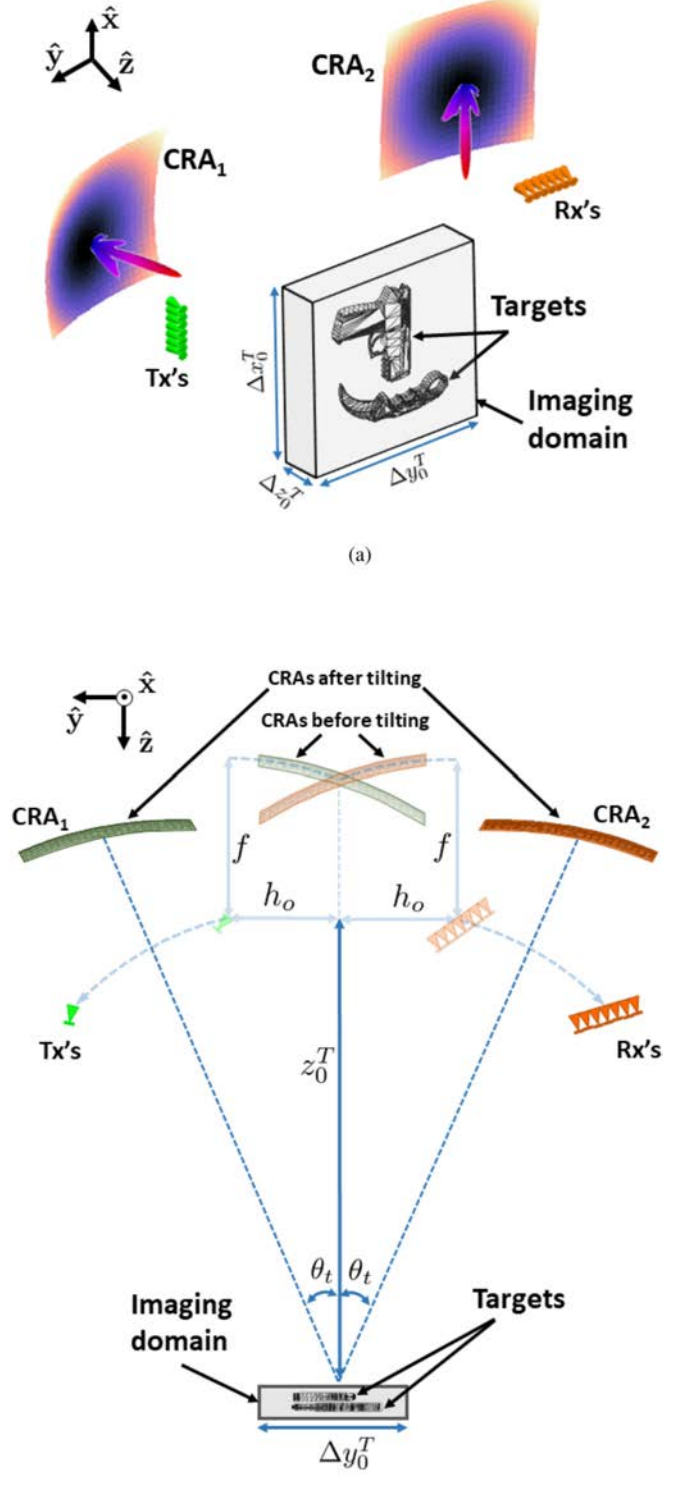


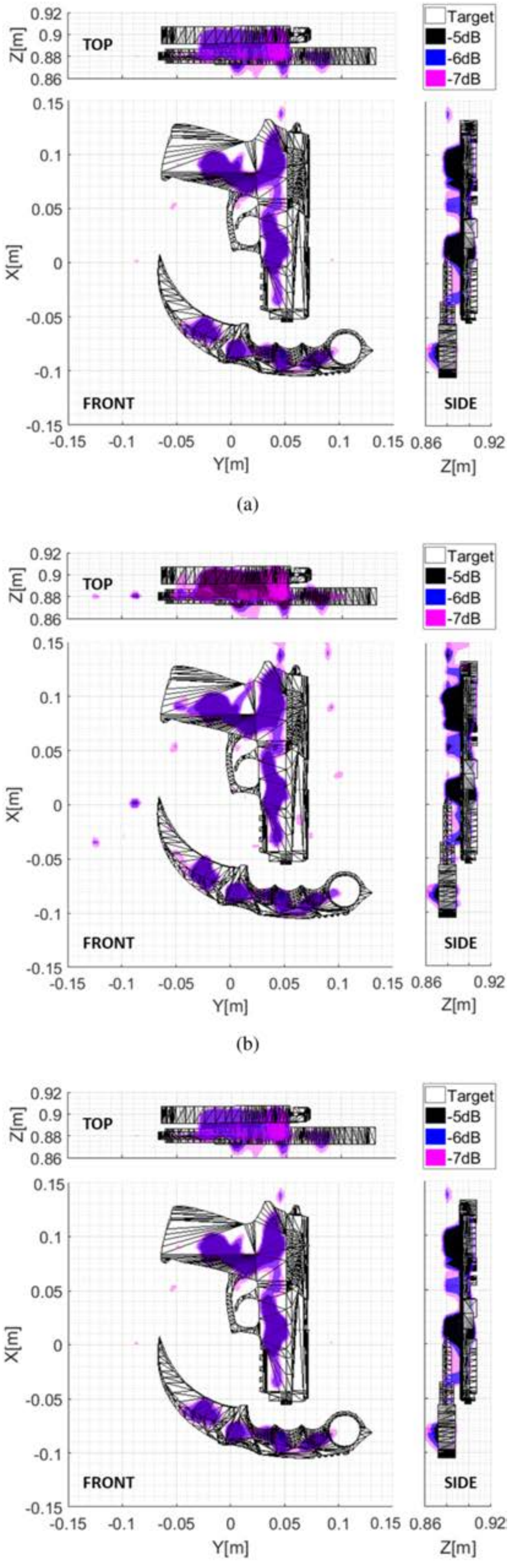
Fig. 10. 2D cross-section of a CRA in offset mode. The scatterers Ω_i distort the phase front creating a pseudo-random pattern.



(a)

(b)

Fig. 11. (a) Geometry of the sensing system. A linear array of transmitters feeds the CRA$_1$, which illuminates the imaging domain. The field scattered by the targets is reflected by the CRA$_2$ and measured by another linear array of receivers, orthogonal to the transmitting one. (b) Top view of the sensing system. The faded CRAs and Tx and Rx arrays indicate their position before tilting. The green CRA (CRA$_1$) is tilted θ_t degrees in the $+\hat{\mathbf{y}}$ direction (counterclockwise), and the orange CRA (CRA$_2$) is tilted θ_t degrees in the $-\hat{\mathbf{y}}$ direction (clockwise).



(a)

(b)

(c)

Fig. 12. Imaging reconstruction (top, front, and side views) using (a) consensus-based ADMM, (b) Sectioning-based ADMM, (c) Consensus and Sectioning-based ADMM. The targets are represented with transparent black triangles and the reconstructed reflectivity is presented in the colored map with the isosurface Matlab command for three levels of the normalized signal.

TABLE IV
METRICS OF THE IMAGE RECONSTRUCTION

	Sensitivity	**Specificity**	**Precision**	**Balanced accuracy**	**F_1−Score**	**$F_{0.5}$−Score**
Consensus-based ADMM	0.3033	0.9926	0.7299	0.6480	0.4286	0.5697
Sectioning-based ADMM	0.3250	0.9886	0.6536	0.6568	0.4341	0.5436
Consensus and Sectioning based ADMM	0.3013	0.9926	0.7285	0.6469	0.4262	0.5675

TABLE V
SUBMATRICES SIZES AND TIME FOR THE THREE ADMM TECHNIQUES

	M	N	**Submatrices sizes**	**Submatrices sizes for inverting with the *matrix inversion lemma***	**Inversion time**	**Convergence time (50 iterations)**	**Imaging time**
Consensus-based ADMM	4	1	540 × 22500	540 × 540	196 ms	616 ms	0.812 s
Sectioning-based ADMM	1	3	2160 × 7500	2160 × 2160	381 ms	639 ms	1.020 s
Consensus and Sectioning based ADMM	4	3	540 × 7500	540 × 540	248 ms	1171 ms	1.419 s

as it is described in [49]. A scatterer is constructed over each patch, with averaged sizes of $\langle D^x \rangle$ and $\langle D^y \rangle$ in the $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ dimensions, respectively. The size in the $\hat{\mathbf{z}}$ dimension D_i^z is defined as the product $\langle D^x \rangle \cdot \tan(\alpha_{t_i})$, with α_{t_i} being the tilt angle for each scatterer, selected from a uniform random variable in the interval $[0, \alpha_{tmax}]$, allowing a maximum tilt angle of α_{tmax}. The material of each scatterer is considered as a perfect electric conductor (PEC), therefore $\sigma_i = \infty$. The imaging domain, where the targets are contained, is located z_0^T meters away from the focal plane of the CRAs before tilting. It covers a parallelepiped-shaped volume defined by the Δx_0^T, Δy_0^T, and Δz_0^T dimensions, and it is discretized into N_p pixels of dimensions l_x, l_y, and l_z. The values for all these parameters are shown in Table III.

Fig. 12 depicts the imaging results when applying the three ADMM techniques for the following parameters: $\rho = 10^5$, $\lambda = 10^{-2}$, and scaling factor $scl = 10^{-4}$ (see Ref. [29]), for 50 iterations and no noise. The targets correspond to a metallic gun and dagger structures located in different planes. Based on the discussion on Section V-B, small values of M and N are selected for the validation of the three techniques. For the consensus-based ADMM, the sensing matrix $\mathbf{H} \in \mathbb{C}^{N_m \times N_p}$ is divided into $M = 4$ submatrices by rows; for the sectioning-based ADMM, $\mathbf{H}$ is divided into $N = 3$ submatrices by columns; and for the consensus and sectioning-based ADMM, $\mathbf{H}$ is divided into $M \cdot N = 4 \cdot 3 = 12$ submatrices by rows and columns.

Table IV indicates six metrics for comparing the imaging reconstruction results for each of the three techniques. A pixel is considered as a detection if its reconstructed normalized signal is equal or above -7 dB. As expected, all values are very similar for each metric. While the sectioning-based ADMM presents a higher sensitivity, namely, it reconstructs the targets better, the precision is lower because of the presence of some artifacts. In terms of the balanced accuracy (mean between sensitivity and specificity) and F_1-score, the sectioning-based ADMM performs slightly better than the other two methods, but it drops when the $F_{0.5}$-score is computed, which gives more importance to the fact of lacking artifacts. Notice that, despite the visualization presents a good reconstruction for the three techniques, the sensitivity values are not very high due to the fact that the targets are not flat and part of the signal may be reflected out of the field of view of the CRA.

Table V shows the sizes of the submatrices for each technique, the inversion time applying the *matrix inversion lemma* for those submatrices, the iterative convergence lapse time, and the total imaging time. The primal and dual residual convergences for each case are shown in Fig. 13.

The times are computed by running an M code in a MATLAB 2017b Parallel Computer Toolbox (PCT); with a GPU Titan V, 5120 CUDA cores (1335 MHz), NVIDIA driver v390.25; in a Ubuntu Linux 16.04.4, kernel 4.13.0-36, operative system. It can

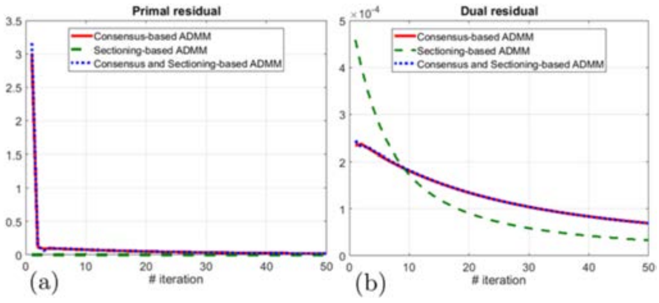


Fig. 13. (a) Primal residual and (b) dual residual of the three ADMM techniques for the imaging example of Fig. 12. The primal residual for the Sectioning-based ADMM is almost zero since there is no *consensus* for this technique.

TABLE VI
AMOUNT OF INFORMATION EXCHANGED PER NODE AT ONE ITERATION

ADMM method	**#of elements to be shared**	**% of element reduction with respect to the consensus-based ADMM**
Consensus-based	$45,000$	0%
Sectioning-based	$6,480$	85.6%
Consensus and Sectioning-based	$12,870$	71.4%

be considered that the three techniques perform the imaging in real time, especially the consensus-based ADMM, since it finds a solution in less than 1 s.

In terms of communication among the computational nodes, Table VI shows the total amount of information that one single node has to exchange at one iteration, for the parameters of this example. It also shows the percentage of shared information reduction for the three techniques, taking the consensus-based ADMM as a reference. It is clear that the sectioning-based ADMM (column-wise division) is the most efficient technique in terms of communication, and the consensus-based ADMM (row-wise division) is the least efficient.

Finally, in order to assess the performance of the imaging in the presence of noise, the reconstructed image of the front view for the Consensus-based ADMM and the $\mathrm{F}_{0.5}$-score are represented in Fig. 14 for different levels of signal to noise ratio, when adding white gaussian noise. The algorithm shows robustness against the noise, as the imaging quality does not degrade much until going below $0\,dB$.

A. Discussion

Comparing the results in terms of imaging quality, imaging time, convergence, and amount of information shared among the computational nodes for the exposed example, none of the three ADMM techniques can be considered the best for all these features. The selection of one or other would depend on the feature of interest or on the physical restriction of the problem. In terms of imaging quality, even though the three techniques perform good imaging, the best option is either consensus- or consensus and sectioning-based ADMM, since they have slightly better performance when promoting the lack of artifacts. In terms of time, consensus-based ADMM has the fastest imaging time; but it is the worst when considering the amount of information exchanged among the nodes. Finally, in terms of convergence and communication efficiency, the sectioning-based ADMM is the winner; however, this method gets slower as the number of divisions gets larger, and the amount of information exchanged increases linearly. Therefore, depending on the particular needs of the problem—accuracy of the imaging, speed, computational nodes architecture, etc.—the selection of one or another method can be contemplated. As a general consideration, the consensus and sectioning-based ADMM technique is always a good option, since it has more degrees of freedom that allow to get close to the best performance for the most of the features.

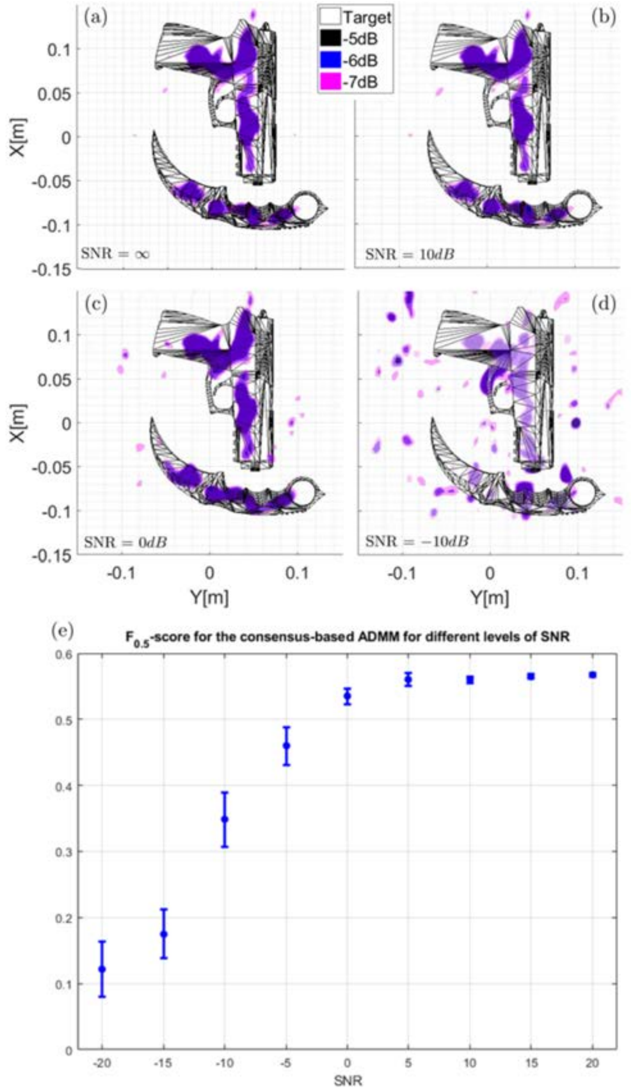


Fig. 14. (Top) Front view imaging reconstruction for the Consensus-based ADMM for different levels of signal to noise ratio: (a) SNR $= \infty$ (No noise), (b) SNR $= 10\,dB$, (c) SNR $= 0\,dB$, (d) SNR $= -10\,dB$, (Bottom) (e) $\mathrm{F}_{0.5}-$score of the reconstructed images with respect to the SNR. The error bars indicate ±1 standard deviation.

It is important to note that larger scale imaging problems would benefit better from these distributed techniques. Low-scale problem might be solved fast enough by a non-distributed method, but when the size of the problem starts to create memory and computational issues to process the imaging, then the partition of the problem would be a necessity, making the use of distributed methods a faster and a more efficient option.

VIII. Conclusion

Three ADMM-based techniques have been introduced to find a sparse solution of a linear matrix equation in a distributed fashion. These techniques are particularly adapted to a mm-wave imaging application. In the *consensus*-based ADMM, the sensing matrix is divided into submatrices by rows, creating several replicas of the unknown imaging vector and solving

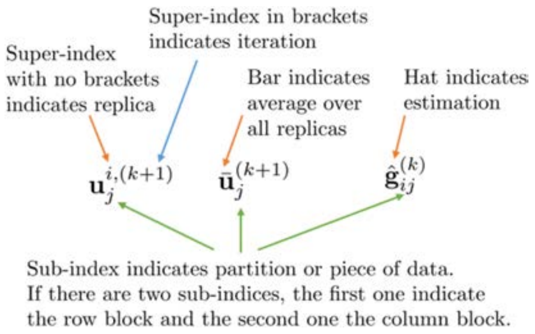

Super-index in brackets indicates iteration
Super-index with no brackets indicates replica
Bar indicates average over all replicas
Hat indicates estimation
$\mathbf{u}_j^{i,(k+1)}$
$\bar{\mathbf{u}}_j^{(k+1)}$
$\hat{\mathbf{g}}_{ij}^{(k)}$
Sub-index indicates partition or piece of data. If there are two sub-indices, the first one indicate the row block and the second one the column block.

Fig. 15. Explanation of the notation during the mathematical development.

them in parallel, reaching a consensus among different solutions and highly accelerating the imaging process. In the *sectioning*-based ADMM, the sensing matrix is divided into submatrices by columns, sectioning the imaging in small regions and optimizing them separately, highly reducing the amount of information exchanged by one node at each iteration. Finally, in the *consensus and sectioning*-based ADMM, the sensing matrix is divided into both rows and columns, segmenting the imaging and creating replicas of each region, combining the advantages of imaging quality and reduced information exchanged among the computational nodes.

A mm-wave imaging example through the use of two CRAs has been presented. The imaging quality defined by several metrics, the imaging time, the convergence, and the communication among the computational nodes have been analyzed and compared. The distributed capabilities of the three proposed techniques have demonstrated their ability of performing real-time imaging of metallic targets with a reduced number of measurements. The application of these methodologies to large-scale imaging problems would allow different degrees of freedom to perform a fast imaging while keeping the communications overhead low.

Imaging structures, such as the CRA, that could further reduce the mutual information among measurements, could diminish the amount of data collected and accelerate the imaging process even more. Besides, convergence analysis and acceleration methods on the proposed techniques may lead to a faster imaging. More decentralized computational architectures can reduce further the amount of information exchanged among the computational nodes. Future analysis will also allow to perform non-regular divisions of the sensing matrix in both rows and columns, in which those divisions may be specified by the user depending on the particular conditions, requirements, and constraints of the problem to be solved. Finally, different types of regularization are also under investigation in order to apply the properties of the distributed ADMM algorithms to more generalized problems, exploring other types of solutions.

Appendix A Notation

As a reference, Fig. 15 explains the meaning of each element in the notation of the variables during the mathematical development.

References

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*. Englewood Cliffs, NJ, USA: Prentice-Hall, Inc., 1989.

[2] D. Greenspan, “Methods of matrix inversion,” *Amer. Math. Monthly*, vol. 62, no. 5, pp. 303–318, 1955.

[3] H. Akaike, “Block Toeplitz matrix inversion,” *SIAM J. Appl. Math.*, vol. 24, no. 2, pp. 234–241, 1973.

[4] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein, “Distributed optimization and statistical learning via the alternating direction method of multipliers,” *Found. Trends Mach. Learn.*, vol. 3, no. 1, pp. 1–122, Jul. 2011.

[5] J. Heredia-Juesas, A. Molaei, L. Tirado, W. Blackwell, and J. Á. Martínez-Lorenzo, “Norm-1 regularized consensus-based admm for imaging with a compressive antenna,” *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 2362–2365, 2017, doi: 10.1109/LAWP.2017.2718242.

[6] P. A. Forero, A. Cano, and G. B. Giannakis, “Consensus-based distributed support vector machines,” *J. Mach. Learn. Res.*, vol. 11, pp. 1663–1707, 2010.

[7] J. F. Mota, J. Xavier, P. M. Aguiar, and M. Puschel, “Distributed basis pursuit,” *IEEE Trans. Signal Process.*, vol. 60, no. 4, pp. 1942–1956, Apr. 2012.

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

[9] A. Olshevsky and J. N. Tsitsiklis, “Convergence speed in distributed consensus and averaging,” *SIAM J. Control Optim.*, vol. 48, no. 1, pp. 33–55, 2009.

[10] M. H. DeGroot, “Reaching a consensus,” *J. Amer. Stat. Assoc.*, vol. 69, no. 345, pp. 118–121, 1974.

[11] L. Fang and P. J. Antsaklis, “On communication requirements for multi-agent consensus seeking,” in *Networked Embedded Sensing and Control*. Berlin/Heidelberg, Germany: Springer, 2006, pp. 53–67.

[12] M. Mehyar, D. Spanos, J. Pongsajapan, S. H. Low, and R. M. Murray, “Distributed averaging on asynchronous communication networks,” in *Proc. 44th IEEE Conf. Decis. Control*, 2005, pp. 7446–7451.

[13] J. F. Mota, J. M. Xavier, P. M. Aguiar, and M. Puschel, “D-ADMM: A communication-efficient distributed algorithm for separable optimization,” *IEEE Trans. Signal Process.*, vol. 61, no. 10, pp. 2718–2723, May 2013.

[14] R. Olfati-Saber, J. A. Fax, and R. M. Murray, “Consensus and cooperation in networked multi-agent systems,” *Proc. IEEE Proc. IRE*, vol. 95, no. 1, pp. 215–233, 2007.

[15] I. D. Schizas, A. Ribeiro, and G. B. Giannakis, “Consensus in ad hoc WSNS with noisy links–Part I: Distributed estimation of deterministic signals,” *IEEE Trans. Signal Process.*, vol. 56, no. 1, pp. 350–364, Jan. 2008.

[16] G. T. Buzzard, S. H. Chan, S. Sreehari, and C. A. Bouman, “Plug-and-play unplugged: Optimization-free reconstruction using consensus equilibrium,” *SIAM J. Imag. Sci.*, vol. 11, no. 3, pp. 2001–2020, 2018.

[17] D. Jakovetic, J. Xavier, and J. M. Moura, “Cooperative convex optimization in networked systems: Augmented Lagrangian algorithms with directed gossip communication,” *IEEE Trans. Signal Process.*, vol. 59, no. 8, pp. 3889–3902, Aug. 2011.

[18] I. D. Schizas, G. B. Giannakis, S. I. Roumeliotis, and A. Ribeiro, “Consensus in ad hoc WSNs with noisy links–Part II: Distributed estimation and smoothing of random signals,” *Signal Process., IEEE Trans.*, vol. 56, no. 4, pp. 1650–1666, Apr. 2008.

[19] G. Oliveri, P. Rocca, and A. Massa, “A Bayesian-compressive-sampling-based inversion for imaging sparse scatterers,” *IEEE Trans. Geosci. Remote Sens.*, vol. 49, no. 10, pp. 3993–4006, Oct. 2011.

[20] M. Aharon, M. Elad, and A. Bruckstein, “K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation,” *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[21] A. Beck and M. Teboulle, “A fast iterative shrinkage-thresholding algorithm for linear inverse problems,” *SIAM J. Imag. Sci.*, vol. 2, no. 1, pp. 183–202, 2009.

[22] U. S. Kamilov, H. Mansour, and B. Wohlberg, “A plug-and-play priors approach for solving nonlinear imaging inverse problems,” *IEEE Signal Process. Lett.*, vol. 24, no. 12, pp. 1872–1876, Dec. 2017.

[23] S. Becker, J. Bobin, and E. J. Candès, “Nesta: A fast and accurate first-order method for sparse recovery,” *SIAM J. Imag. Sci.*, vol. 4, no. 1, pp. 1–39, 2011.

[24] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2009.

[25] J. Heredia Juesas, G. Allan, A. Molaei, L. Tirado, W. Blackwell, and J. A. Martinez Lorenzo, "Consensus-based imaging using ADMM for a compressive reflector antenna," in *Proc. Int. Symp. Antennas Propag. Symp.*, 2015, pp. 1304–1305.

[26] T. Erseghe, D. Zennaro, E. Dall'Anese, and L. Vangelista, "Fast consensus by the alternating direction multipliers method," *IEEE Trans. Signal Process.*, vol. 59, no. 11, pp. 5523–5537, Nov. 2011.

[27] E. J. Candes and T. Tao, "Decoding by linear programming," *IEEE Trans. Inf. Theory*, vol. 51, no. 12, pp. 4203–4215, 2005.

[28] C. Ramirez, V. Kreinovich, and M. Argaez, "Why *l*1 is a good approximation to *l*0: A geometric explanation," *J. Uncertain Syst.*, vol. 7, no. 3, pp. 203–207, 2013.

[29] J. Heredia-Juesas, A. Molaei, L. Tirado, and J. A. Martinez-Lorenzo, "Sectioning-based ADMM imaging for fast node communication with a compressive antenna," *IEEE Antennas Wireless Propag. Lett.*, vol. 18, no. 2, pp. 226–230, Feb. 2018.

[30] J. Heredia-Juesas, A. Molaei, L. Tirado, and J. Á. Martínez-Lorenzo, "Fast node communication ADMM-based imaging algorithm with a compressive reflector antenna," in *Proc. Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, 2018, pp. 1304–1305.

[31] J. Heredia-Juesas, L. Tirado, A. Molaei, and J. A. Martinez-Lorenzo, "ADMM based consensus and sectioning norm-1 regularized algorithm for imaging with a CRA," in *Proc. IEEE Int. Symp. Antennas Propag. USNC-URSI Radio Sci. Meeting*, 2019, pp. 549–550.

[32] A. Molaei, J. H. Juesas, and J. A. M. Lorenzo, "Compressive reflector antenna phased array," in *Antenna Arrays and Beam-Formation*. London, U.K.: InTech, 2017.

[33] J. Martinez Lorenzo, J. HerediaJuesas, and W. Blackwell, "A single-transceiver compressive reflector antenna for high-sensing-capacity imaging," *IEEE Antennas Wireless Propag. Lett.*, vol. 15, pp. 968–971, Mar. 2016, doi: 10.1109/LAWP.2015.2487319.

[34] J. Martinez Lorenzo, J. HerediaJuesas, and W. Blackwell, "Single-transceiver compressive antenna for high-capacity sensing and imaging applications," in *Proc. 9th Eur. Conf. Antennas Propag.*, 2015, pp. 1–3.

[35] M. V. Afonso, J. M. Bioucas-Dias, and M. A. Figueiredo, "Fast image recovery using variable splitting and constrained optimization," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2345–2356, Sep. 2010.

[36] D. P. Bertsekas, *Constrained Optimization and Lagrange Multiplier Methods*. Cambridge, MA, USA: Academic Press, 2014.

[37] E. J. Candes, "The restricted isometry property and its implications for compressed sensing," *Comptes Rendus Mathematique*, vol. 346, no. 9-10, pp. 589–592, 2008.

[38] R. Obermeier and J. A. Martinez-Lorenzo, "Model-based optimization of compressive antennas for high-sensing-capacity applications," *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 1123–1126, 2017, doi: 10.1109/LAWP.2016.2623789.

[39] K. Bredies and D. A. Lorenz, "Linear convergence of iterative soft-thresholding," *J. Fourier Anal. Appl.*, vol. 14, no. 5-6, pp. 813–837, Oct. 2008.

[40] M. A. Woodbury, "Inverting modified matrices," *Memorandum Rep.*, vol. 42, Statistical Research Group, Princeton University, Princeton, NJ, 1950.

[41] A. Molaei, J. H. Juesas, W. Blackwell, and J. A. M. Lorenzo, "Interferometric sounding using a metamaterial-based compressive reflector antenna," *IEEE Trans. Antennas Propag.*, vol. 66, no. 5, pp. 2188–2198, May 2018, doi: 10.1109/TAP.2018.2809488.

[42] A. Molaei, G. Allan, J. Heredia, W. Blackwell, and J. Martinez-Lorenzo, "Interferometric sounding using a compressive reflector antenna," in *Proc. 10th Eur. Conf. Antennas Propag.*, 2016, pp. 1–4.

[43] A. Molaei, J. Heredia-Juesas, and J. Martinez-Lorenzo, "A 2-bit and 3-bit metamaterial absorber-based compressive reflector antenna for high sensing capacity imaging," in *Proc. IEEE Int. Symp. Technol. Homeland Secur.*, 2017, pp. 1–6.

[44] A. Molaei, J. H. Juesas, G. Allan, and J. Martinez-Lorenzo, "Active imaging using a metamaterial-based compressive reflector antenna," in *Proc. IEEE Int. Symp. Antennas Propag.*, 2016, pp. 1933–1934.

[45] A. Molaei, G. Ghazi, J. Heredia-Juesas, H. Gomez-Sousa, and J. Martinez-Lorenzo, "High capacity imaging using an array of compressive reflector antennas," in *Proc. 11th Eur. Conf. Antennas Propag.*, 2017, pp. 1731–1734.

[46] A. Molaei, J. H. Juesas, and J. Martinez-Lorenzo, "Single-pixel mm-wave imaging using 8-bits metamaterial-based compressive reflector antenna," in *Proc. Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, 2017, pp. 847–848.

[47] A. Molaei *et al.*, "Experimental results of a compressive reflector antenna producing spatial coding," in *Proc. Int. Symp. Antennas Propag. USNC/URSI Nat. Radio Sci. Meeting*, 2018, pp. 933–934.

[48] R. Obermeier and J. A. Martinez-Lorenzo, "Model-based optimization of compressive antennas for high-sensing-capacity applications," *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 1123–1126, 2017, doi: 10.1109/LAWP.2016.2623789.

[49] J. Meana, J. Martinez-Lorenzo, F. Las-Heras, and C. Rappaport, "Wave scattering by dielectric and lossy materials using the modified equivalent current approximation (meca)," *IEEE Trans. Antennas Propag.*, vol. 58, no. 11, pp. 3757–3761, Nov. 2010.