IEEE TRANSACTIONS ON MEDICAL IMAGING, VOL. 30, NO. 3, MARCH 2011

Parallel MR Image Reconstruction Using Augmented Lagrangian Methods

Sathish Ramani*, *Member, IEEE*, and Jeffrey A. Fessler, *Fellow, IEEE*

Abstract

SENSE needs regularization to noise and aliasing suppression

Ex. Edge preserving, sparsity-based regularization

-> computationally intensive(non-linear optimization)

Augmented Lagrangian (AL) framework

- for solving large-scale constrained optimization problems.

1. Formulate regularized SENSE as an unconstrained optimization task.
2. Convert it to a set of constrained problems using **variable splitting**.
3. Solve it using **alternating minimization** method.

AL algorithms converge faster than general-purpose optimization algorithms (NCG, MFISTA)

1. Introduction

2. Problem Formulation (P0)

3. Constrained Optimization and Augmented Lagrangian (AL) Formalism

4. Proposed AL Algorithms for Regularized SENSE-Reconstruction

A. Splitting the regularization term (P1)

1) Minimization with respect to x
2) Minimization with respect to u_1
3) Several cases
4) AL algorithm for Problem P1

B. Splitting the Fourier encoding and spatial components in the data-fidelity term (P2)

1) Minimization with respect to $\mathbf{u}_{0,2}$ and x
2) Implementing the matrix inverse
3) AL algorithm for Problem P2

C. Choosing μ- and ν- values for the AL algorithms

1.
Introduction

1. Introduction

SENSE is a pMRI technique where reconstruction is performed by **solving a linear system** that explicitly depends on the **sensitivity maps**.

However, reconstruction inherently suffer from **SNR degradation** in the presence of **noise** mainly due to k-space **undersampling** and instability arising from **correlation in sensitivity maps**.

Regularization is an attractive means of **restoring stability** where prior information can also be incorporated effectively.

We formulate **regularized SENSE-reconstruction** as an unconstrained optimization problem.

1. Introduction

This paper presents accelerated algorithms for **regularized SENSE**-reconstruction using **the augmented Lagrangian (AL) formalism**.

To use the AL formalism, we first **convert the unconstrained** problem in to an equivalent **constrained** optimization problem using a technique called ***variable splitting*** where **auxiliary variables** take the place of linear transformations of X in the cost function J.

Then, we construct a corresponding AL function and **minimize it alternatively with respect to one auxiliary variable** at a time—this step forms the key ingredient as it decouples the minimization process and simplifies optimization

We also propose to use a **diagonal weighting term** in the AL formalism to induce suitable **balance between various constraints** because the matrix-elements associated with Fourier encoding and the sensitivity maps can be of different orders of magnitude in SENSE.

2. Problem formulation

Section II formulates the regularized SENSE-reconstruction problem (with sparsity-based regularization) as an unconstrained optimization task.

2. Problem Formulation

SENSE MR imaging model : $$\mathbf{d} = \mathbf{F}\,\mathbf{S}\,\mathbf{x} + \boldsymbol{\varepsilon} \qquad (1)$$

$\boldsymbol{x}$: *unknown image to be reconstructed*
$[N \; X \; 1]$

$\mathbf{d}$: *sampling data(kspace)*
$[ML \; X \; 1]$

$\boldsymbol{\varepsilon}$: *noise*
$[ML \; X \; 1]$

$\boldsymbol{S}$: *sensitivity map*
$[NL \; X \; N]$, $S = [S_1; \ldots; S_L], S_l : [N \; X \; N]$ *diagonal matrix*

$\mathbf{F}$: *FourierUndersampling encoding matrix*
$[ML \; X \; NL]$, $F = I_L \otimes F_u, F_u : [M \; X \; N]$ *Fourier encoding matrix*, I_L: *identity matrix of size L*

Since **regularization** is able to reducing aliasing artifacts and the effect of noise(by incorporating prior knowledge), we formulate the problem in a penalized-likelihood setting and solve it by minimizing a cost criterion

$$\mathbf{P0} : \hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \left\{ J(\mathbf{x}) = \frac{1}{2} \|\mathbf{d} - \mathbf{F}\,\mathbf{S}\,\mathbf{x}\|^2_{\mathbf{K}_{ML}} + \Psi(\mathbf{x}) \right\} \qquad (2)$$

$\mathrm{K_{ML}}$: *inverse of the noise covariance matrix*
$[ML \; X \; ML]$, $\mathrm{K_{ML}} = \mathrm{K_S} \otimes \mathrm{I_M}$, $\mathrm{K_S}$: [L X L] inverse of the covariance matrix from L coils ; $||\mathrm{u}||^2_{\mathrm{K}} = u^H K u$

The weighting matrix $\mathrm{K_{ML}}$ can be eliminated from cost function by applying a **noise-decorrelation** procedure.

// $\boldsymbol{K_{ML}}$ assumes the fact that noise from different coils may be correlated only over space (i.e., coils) and not over k-space.

2. Problem Formulation

$$\mathbf{P0}:\hat{\mathbf{x}}=\arg\min_{\mathbf{x}}\left\{J(\mathbf{x})=\underbrace{\frac{1}{2}\|\mathbf{d}-\mathbf{F}\,\mathbf{S}\,\mathbf{x}\|^2_{\mathbf{K}_{ML}}}_{①}+\underbrace{\Psi(\mathbf{x})}_{②}\right\} \quad (2)$$

① : Noise decorrelation

$$\mathbf{K}_s=\tilde{\mathbf{K}}_s^{\mathrm{H}}\tilde{\mathbf{K}}_s \qquad \text{// Cholesky decomposition}$$

$$\mathbf{K}_{ML}=\tilde{\mathbf{K}}_{ML}^{\mathrm{H}}\tilde{\mathbf{K}}_{ML} \quad \text{where } \tilde{\mathbf{K}}_{ML}=\tilde{\mathbf{K}}_s\otimes\mathbf{I}_M \qquad // (\boldsymbol{A}\otimes\boldsymbol{B})^{\boldsymbol{H}}=(\boldsymbol{A}^{\boldsymbol{H}}\otimes\boldsymbol{B}^{\boldsymbol{H}})$$

$$\begin{aligned}\tilde{\mathbf{K}}_{ML}\mathbf{F}&=(\tilde{\mathbf{K}}_s\otimes\mathbf{I}_M)(\mathbf{I}_L\otimes\mathbf{F}_u)=(\tilde{\mathbf{K}}_s\mathbf{I}_L\otimes\mathbf{I}_M\mathbf{F}_u)\\&=(\mathbf{I}_L\tilde{\mathbf{K}}_s\otimes\mathbf{F}_u\mathbf{I}_N)=(\mathbf{I}_L\otimes\mathbf{F}_u)(\tilde{\mathbf{K}}_s\otimes\mathbf{I}_N)=\mathbf{F}\tilde{\mathbf{K}}_{NL}\end{aligned}$$

$$//(\boldsymbol{A}\otimes\boldsymbol{B})(\boldsymbol{C}\otimes\boldsymbol{D})=(\boldsymbol{AC})\otimes(\boldsymbol{BD})$$

where $\tilde{\mathbf{K}}_{NL}\triangleq\tilde{\mathbf{K}}_s\otimes\mathbf{I}_N$. Letting

$$\tilde{\mathbf{d}}\triangleq\tilde{\mathbf{K}}_{ML}\mathbf{d} \qquad (3)$$

$$\tilde{\mathbf{S}}\triangleq\tilde{\mathbf{K}}_{NL}\mathbf{S} \qquad (4)$$

we therefore get that

$$\frac{1}{2}\|\mathbf{d}-\mathbf{F}\,\mathbf{S}\,\mathbf{x}\|^2_{\mathbf{K}_{ML}}=\frac{1}{2}\|\tilde{\mathbf{d}}-\mathbf{F}\,\tilde{\mathbf{S}}\mathbf{x}\|_2^2 \qquad (5)$$

② : General expression of regularizor

$$\Psi(\mathbf{x})=\sum_{q=1}^{Q}\lambda_q\sum_{n=1}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{R}_{pq}\mathbf{x}]_n|^{m_q}\right) \qquad (6)$$

$\boldsymbol{q}$ → *regularizer*
$\boldsymbol{p}$ → *regularizer's subordinate matrix* (property)
Φ → *some operator which complete the regularizer*

Examples)

1) ℓ_1-norm of wavelet coefficients: $Q=1$, $P_1=1$, $m_1=1$, $\mathbf{R}_{11}=\mathbf{W}$ is a wavelet transform (orthonormal or a tight frame), and $\Phi_{1n}(x)=x$ where n indexes the rows of $\mathbf{R}_{11}$.
2) Discrete isotropic total-variation (TV) regularization [15]: $Q=1$, $P_1=2$, $m_1=2$, $\mathbf{R}_{11}$ and $\mathbf{R}_{21}$ represent horizontal and vertical finite-differencing matrices, respectively, and $\Phi_{1n}(x)=\sqrt{x}$, where n indexes the rows of $\mathbf{R}_{\cdot 1}$.
3) Discrete anisotropic total-variation (TV) regularization [15]: $Q=1$, $P_1=2$, $m_1=1$, $\mathbf{R}_{11}$ and $\mathbf{R}_{21}$ represent horizontal and vertical finite-differencing matrices, respectively, and $\Phi_{1n}(x)=x$, where n indexes rows of $\mathbf{R}_{\cdot 1}$.

2. Problem Formulation

$$\mathbf{P0}:\hat{\mathbf{x}}=\arg\min_{\mathbf{x}}\left\{J(\mathbf{x})=\frac{1}{2}\|\mathbf{d}-\mathbf{F}\,\mathbf{S}\,\mathbf{x}\|^2_{\mathbf{K}_{ML}}+\Psi(\mathbf{x})\right\} \quad (2)$$

$$=\arg\min_{\mathbf{x}}\left\{J(\mathbf{x})=\frac{1}{2}\|\tilde{\mathbf{d}}-\mathbf{F}\,\tilde{\mathbf{S}}\mathbf{x}\|^2_2+\sum_{q=1}^{Q}\lambda_q\sum_{n=1}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{R}_{pq}\,\mathbf{x}]_n|^{m_q}\right)\right\}$$

The basic idea is to break down **P0** in to smaller tasks by introducing “**artificial**” constraints.

That are designed so that the **subproblems become decoupled** and can be solved relatively rapidly.

3.
Constrained Optimization and Augmented Lagrangian (AL) Formalism

Section III presents a quick overview of AL framework.

3. Constrained Optimization and Augmented Lagrangian (AL) Formalism

Constrained optimization problem

$$\hat{\mathbf{u}} = \arg\min_{\mathbf{u}\in\Omega^N} f(\mathbf{u}) \ \text{ subject to } \ \mathbf{C}\mathbf{u} = \mathbf{b} \qquad (7)$$

where Ω is $\mathbb{R}$ or $\mathbb{C}$,
f is a real convex function
$\mathbf{C}$ is a $M \times N$ matrix that specifies the constraint equations
$\mathbf{b} \in \Omega^M$

Augmented Lagrangian function (Unconstrained optimization)

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}, \mu) = f(\mathbf{u}) + \boldsymbol{\gamma}^{\mathrm{H}}(\mathbf{C}\mathbf{u} - \mathbf{b}) + \frac{\mu}{2}||\mathbf{C}\mathbf{u} - \mathbf{b}||_2^2 \qquad (8)$$

// Augmented (penalty) term

where $\boldsymbol{\gamma} \in \Omega^M$ the vector of Lagrange multipliers
$\mu > 0$.

- **Solver :** alternative minimization w.r.t. $\boldsymbol{u}$ for a fixed $\boldsymbol{\mu}$(penalty weight) and updating $\boldsymbol{\gamma}$(Lagrange factor)

$$\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}} \mathcal{L}(\mathbf{u}, \boldsymbol{\gamma}^{(j)}, \mu), \qquad (9)$$

$$\boldsymbol{\gamma}^{(j+1)} = \boldsymbol{\gamma}^{(j)} + \mu(\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b}) \qquad (10)$$

$\boxed{\text{An important aspect of the AL scheme is that convergence may be guaranteed without the need for changing } \mu}$

3. Constrained Optimization and Augmented Lagrangian (AL) Formalism

The AL function L can be rewritten by grouping together the terms involving $\boldsymbol{Cu} - \boldsymbol{b}$ as

$$\mathcal{L}(\mathbf{u}, \boldsymbol{\eta}, \mu) = f(\mathbf{u}) + \frac{\mu}{2}\|\mathbf{Cu} - \boldsymbol{\eta}\|_2^2 + C_{\boldsymbol{\gamma}} \qquad (13)$$

where $\boldsymbol{\eta} \triangleq \mathbf{b} - (1/\mu)\boldsymbol{\gamma}$

$C_{\boldsymbol{\gamma}}$ is a constant independent of $\mathbf{u}$

Algorithm AL

1. Select $\mathbf{u}^{(0)}$, $\boldsymbol{\eta}^{(0)}$, and $\mu > 0$; set $j = 0$

Repeat

2. $\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}}\{f(\mathbf{u}) + (\mu/2)\|\mathbf{Cu} - \boldsymbol{\eta}^{(j)}\|_2^2\}$
3. $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{Cu}^{(j+1)} - \mathbf{b})$
4. Set $j = j + 1$

Until stop-criterion is met

Bregman iterations

$$\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}} D_f(\mathbf{u}, \mathbf{u}^{(j)}, \boldsymbol{\rho}^{(j)}) + \frac{\mu}{2}\|\mathbf{C}\mathbf{u} - \mathbf{b}\|_2^2 \quad (11)$$

$$\boldsymbol{\rho}^{(j+1)} = \boldsymbol{\rho}^{(j)} - \mu\mathbf{C}^{\mathrm{H}}(\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b}) \quad (12)$$

where $D_f(\mathbf{u}, \mathbf{v}, \boldsymbol{\rho}) = f(\mathbf{u}) - f(\mathbf{v}) - \boldsymbol{\rho}^{\mathrm{H}}(\mathbf{u} - \mathbf{v})$ // Bregman distance

$\boldsymbol{\rho}$ is a $N \times 1$ vector in the subgradient of f at $\mathbf{u}$.

If $\boldsymbol{\rho} = -\boldsymbol{C}^{H}\boldsymbol{\gamma}$ then, $D_f\left(\boldsymbol{u}, \boldsymbol{u}^{(j)}, \boldsymbol{\rho}^{(j)}\right) + \frac{\mu}{2}\left|\left|\boldsymbol{C}\boldsymbol{u} - \boldsymbol{b}\right|\right|_2^2$ is identical to $£\,(\boldsymbol{u}, \boldsymbol{\gamma}^{(j)}, \mu)$ // up to constants irrelevant for optimization

The Bregman iterations converge exactly under mentioned condition

-> Step 2 of the AL algorithm will use this property

However, this step may be computationally expensive and is often replaced by an inexact minimization.

Numerical evidence suggests that inexact minimizations can still be effective in the Bregman/AL scheme.

4.
Proposed AL Algorithms for Regularized SENSE-Reconstruction

Algorithm AL

1. Select $\mathbf{u}^{(0)}$, $\boldsymbol{\eta}^{(0)}$, and $\mu > 0$; set $j = 0$

Repeat

2. $\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}}\{f(\mathbf{u}) + (\mu/2)\|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}^{(j)}\|_2^2\}$

3. $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b})$

4. Set $j = j + 1$

Until stop-criterion is met

// How to solve this part?

Section IV applies the AL formalism to regularized SENSE-reconstruction in detail.

Variable Splitting with auxiliary variables

Unconstrained problem :

$$\min_{\mathbf{u}\in\mathbb{R}^n} f_1(\mathbf{u}) + f_2\left(g(\mathbf{u})\right),$$

Constrained problem :

$$\begin{array}{ll} \min\limits_{\mathbf{u}\in\mathbb{R}^n,\, \mathbf{v}\in\mathbb{R}^d} & f_1(\mathbf{u}) + f_2(\mathbf{v}) \\ \text{subject to} & g(\mathbf{u}) = \mathbf{v}, \end{array}$$

Auxiliary variable

In this paper,

$$f_1(u) + f_2\big(g(u)\big) = \boldsymbol{P0}$$

$$\mathbf{P0} := \arg\min_{\mathbf{x}} \left\{ J(\mathbf{x}) = \frac{1}{2}\|\tilde{\mathbf{d}} - \mathbf{F}\,\tilde{\mathbf{S}}\mathbf{x}\|_2^2 + \sum_{q=1}^{Q} \lambda_q \sum_{n=1}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{R}_{pq}\,\mathbf{x}]_n|^{m_q} \right) \right\}$$

$$g(u) : linear\ transformation(FSx, Rx)$$

A. Splitting the Regularization Term

The resulting constrained formulation of **P0** is

$$\mathbf{P1}: \min_{\mathbf{u}_1,\mathbf{x}} J_1(\mathbf{x},\mathbf{u}_1) \text{ subject to } \mathbf{u}_1 = \mathbf{R}\mathbf{x}$$

where

$$J_1(\mathbf{x},\mathbf{u}_1) \triangleq \frac{1}{2}\|\mathbf{d}-\mathbf{F}\mathbf{S}\mathbf{x}\|^2 + \sum_{q=1}^{Q}\lambda_q \sum_{n=1}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{u}_{1pq}]_n|^{m_q}\right)$$

$\mathbf{u}_{1pq} = \mathbf{R}_{pq}\mathbf{x},\ p = 1,\ldots,P_q\ \forall\, q$

In AL framework,

$$\mathcal{L}_1(\mathbf{u},\boldsymbol{\gamma}_1,\mu) = J_1(\mathbf{x},\mathbf{u}_1) + \boldsymbol{\gamma}_1^{\mathrm{H}}\mathbf{C}\mathbf{u} + \frac{\mu}{2}\|\mathbf{C}\mathbf{u}\|_2^2.$$

$\mathbf{u} = \begin{bmatrix}\mathbf{u}_1\\ \mathbf{x}\end{bmatrix},\ f(\mathbf{u}) = J_1(\mathbf{x},\mathbf{u}_1),\ \mathbf{C} = [\mathbf{I}_R\ -\mathbf{R}],\ \mathbf{b} = \mathbf{0}.$

After ignoring irrelevant constants,

$$\mathcal{L}_1(\mathbf{u},\boldsymbol{\eta}_1,\mu) = J_1(\mathbf{x},\mathbf{u}_1) + \frac{\mu}{2}\|\mathbf{C}\mathbf{u}-\boldsymbol{\eta}_1\|^2 \qquad (14)$$

where $\boldsymbol{\eta}_1 = -(1/\mu)\boldsymbol{\gamma}_1$

A. Splitting the Regularization Term

Solve it by using **alternative minimization** method

cf) $\mathcal{L}_1(\mathbf{u}, \boldsymbol{\eta}_1, \mu) = \frac{1}{2}\|\mathbf{d} - \mathbf{FSx}\|^2 + \sum_{q=1}^{Q} \lambda_q \sum_{n=1}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q}\right) + \frac{\mu}{2}\|\mathbf{Cu} - \boldsymbol{\eta}_1\|^2$

$$\mathbf{u}_1^{(j+1)} = \arg\min_{\mathbf{u}_1} \mathcal{L}_1(\mathbf{u}_1, \mathbf{x}^{(j)}, \boldsymbol{\eta}_1^{(j)}, \mu) \qquad (15)$$

$$\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}} \mathcal{L}_1(\mathbf{u}_1^{(j+1)}, \mathbf{x}, \boldsymbol{\eta}_1^{(j)}, \mu). \qquad (16)$$

1) Minimization w.r.t. x // eq. 16

2) Minimization w.r.t. u_1 // eq. 15

A.1. Minimization w.r.t. x

cf) $\mathcal{L}_1(\mathbf{u}, \boldsymbol{\eta}_1, \mu) = \frac{1}{2}\|\mathbf{d} - \mathbf{FSx}\|^2 + \sum_{q=1}^{Q} \lambda_q \sum_{n=1}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q}\right) + \frac{\mu}{2}\|\mathbf{Cu} - \boldsymbol{\eta}_1\|^2$

Minimization w.r.t. x

$$\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}} \mathcal{L}_1(\mathbf{u}_1^{(j+1)}, \mathbf{x}, \boldsymbol{\eta}_1^{(j)}, \mu).$$

$$= \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{d} - \mathbf{FSx}\|_2^2 + \frac{\mu}{2}\left\|\mathbf{u}_1^{(j+1)} - \mathbf{Rx} - \boldsymbol{\eta}_1^{(j)}\right\|_2^2 \right\}$$

// constants are omitted

$$= \mathbf{G}_\mu^{-1}[\mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu\mathbf{R}^{\mathrm{H}}(\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_1^{(j)})] \quad (17)$$

where

$$\mathbf{G}_\mu = \mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{FS} + \mu\mathbf{R}^{\mathrm{H}}\mathbf{R}. \quad (18)$$

Computing $\boldsymbol{G}_{\boldsymbol{\mu}}^{-1}$ is impractical for large N.

Therefore, few iterations of CG algorithm will be applied (CG is precalculated before AL iteration)

// no iteration affected
variables in G^-1

A.2. Minimization w.r.t. $\mathbf{u}_1$

cf) $\mathcal{L}_1(\mathbf{u}, \boldsymbol{\eta}_1, \mu) = \frac{1}{2}\|\mathbf{d} - \mathbf{FSx}\|^2 + \sum_{q=1}^{Q} \lambda_q \sum_{n=1}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q}\right) + \frac{\mu}{2}\|\mathbf{Cu} - \boldsymbol{\eta}_1\|^2$

Minimization w.r.t. $\mathbf{u}_1$

$$
\begin{aligned}
\mathbf{u}_1^{(j+1)} &= \arg\min_{\mathbf{u}_1} \mathcal{L}_1(\mathbf{u}_1, \mathbf{x}^{(j)}, \boldsymbol{\eta}_1^{(j)}, \mu) \\
&= \arg\min_{\mathbf{u}_1} \left\{ \begin{array}{l} \sum_{q=1}^{Q} \lambda_q \sum_{n=1}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q}\right) \\ + \frac{\mu}{2}\left\|\mathbf{u}_1 - \mathbf{Rx}^{(j)} - \boldsymbol{\eta}_1^{(j)}\right\|_2^2 \end{array} \right\}. \qquad (19)
\end{aligned}
$$

// constants are omitted

Let's decompose this large-scale problem into smaller minimization tasks

Let $\mathbf{r}^{(j)} = \mathbf{Rx}^{(j)}$; for each q and n, we collect $\mathbf{v}_{qn} = \{u_{1pqn}\}_{p=1}^{P_q}$, $\boldsymbol{\varrho}_{qn}^{(j)} = \{r_{pqn}^{(j)}\}_{p=1}^{P_q}$, and $\boldsymbol{\beta}_{qn}^{(j)} = \{\eta_{1pqn}^{(j)}\}_{p=1}^{P_q}$, so that $\mathbf{v}_{qn}, \boldsymbol{\varrho}_{qn}^{(j)}, \boldsymbol{\beta}_{qn}^{(j)} \in \mathbb{C}^{P_q}$.

Minimization w.r.t. $\mathbf{u_1}$

$$\mathbf{v}_{qn}^{(j+1)} = \arg\min_{\mathbf{v}_{qn}} \left\{ \frac{\lambda_q}{\mu} \Phi_{qn}\left(\|\mathbf{v}_{qn}\|_{m_q}^{m_q} \right) + \frac{1}{2} \left\| \mathbf{v}_{qn} - (\boldsymbol{\varrho}_{qn}^{(j)} + \boldsymbol{\beta}_{qn}^{(j)}) \right\|_2^2 \right\}. \quad (20)$$

Gradient :

$$(\boldsymbol{\Theta}(\mathbf{v}) + \mathbf{I}_P)\mathbf{v}|_{\mathbf{v}=\mathbf{v}^{(j+1)}} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)} \quad (21)$$

where $\boldsymbol{\Theta}(\mathbf{v}) \triangleq \mathrm{diag}\left\{ \frac{\lambda m}{\mu} \Phi'(\|\mathbf{v}\|_m^m)|v_k|^{m-2} \right\}$ (22)

The main obstacles to obtaining a direct solution of (20) are the coupling introduced between different components of **v** **[**$\boldsymbol{\Phi}'(\|\boldsymbol{v}\|_m^m)$ and the presence of the $|\boldsymbol{v}_k|^{m-2}$ in $\boldsymbol{\Theta}(\boldsymbol{v})$**].**

A.3. Several cases

The main obstacles can be circumvented and here's some cases

1) l1-regularization

- m=1, $\Phi(x) = x$

Gradient : $\left(\frac{\lambda}{\mu|v_k|}+1\right)v_k = \varrho_k^{(j)} + \beta_k^{(j)}$

where $v_k \neq 0$, $k = 1, 2, \ldots P$.

By shrinkage rule,

$$v_k^{(j+1)} = \text{shrink}\left\{\varrho_k^{(j)} + \beta_k^{(j)}, \frac{\lambda}{\mu}\right\} \ \forall\, k,$$

where $\text{shrink}\{d, \lambda\} = (d/|d|)\max\{|d| - \lambda, 0\}$.

2) P=1 // P : size of 'v'

- 1D minimization => achieved numerically for a general Φ or analytically for m=1

A.3. Several cases

The main obstacles can be circumvented and here's some cases

3) m=2 and A general Φ

Gradient : $\underbrace{\left(\frac{2\lambda}{\mu}\Phi'(||\mathbf{v}||_2^2)+1\right)}_{\chi(||\mathbf{v}||_2)}\mathbf{v} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}.$

Solve $\mathrm{X}(||v||_2)$ numerically by using a look-up table for Φ' to find the value for the shrinkage factor $\mathrm{X}(||v||_2)$

In summary, the minimization problem(20) is fairly simple and fast typically.

4) TV-type regularization

- m=2, $\Phi(x) = \sqrt{x}$

Gradient : $\chi(||\mathbf{v}||_2)\mathbf{v} = \boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}$ (24)

where $\chi(||\mathbf{v}||_2) = ((\lambda/\mu||\mathbf{v}||_2) + 1)$ for $\mathbf{v} \neq \mathbf{0}$.

Taking l2-nrom on both sides and manipulating,

$$||\mathbf{v}||_2|_{\mathbf{v}=\mathbf{v}^{(j+1)}} = ||\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}||_2 - \lambda/\mu,$$

$$\chi(||\mathbf{v}^{(j+1)}||_2) = \left(\frac{||\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}||_2}{||\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}||_2 - \frac{\lambda}{\mu}}\right)$$

With (24), upper equation and shrinkage rule,

$$\mathbf{v}^{(j+1)} = \mathrm{shrink}_{\mathrm{vec}}\left\{\boldsymbol{\varrho}^{(j)} + \boldsymbol{\beta}^{(j)}, \frac{\lambda}{\mu}\right\}$$

where $\mathrm{shrink}_{\mathrm{vec}}\{\mathbf{d}, \lambda\} \triangleq (\mathbf{d}/||\mathbf{d}||_2)\max\{||\mathbf{d}||_2 - \lambda, 0\}$

AL algorithm for problem P1

AL-P1: AL Algorithm for solving problem P1

1. Select $\mathbf{x}^{(0)}$ and $\mu > 0$
2. Precompute $\mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{d}$; set $\boldsymbol{\eta}_1^{(0)} = \mathbf{0}$ and $j = 0$

Repeat:

3. Obtain an update $\mathbf{u}_1^{(j+1)}$ using an appropriate technique as described in **Sections IV-A2 to IV-A6** // cases being shown
4. Obtain an update $\mathbf{x}^{(j+1)}$ by running few CG iterations on **(17)** // most complex part(CG)
5. $\boldsymbol{\eta}_1^{(j+1)} = \boldsymbol{\eta}_1^{(j)} - (\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{x}^{(j+1)})$
6. Set $j = j + 1$

Until stop-criterion is met

Lagrange multiplier method

AL

penalty

PO : $\hat{x} = \arg\min_x \{J(x)\}$ where $J(x) = f(x) + \varphi(x)$

$= \frac{1}{2}\|d - FSx\|_2^2 + \sum \lambda \sum \phi(\sum [Rx])$

(Unconstrained)

(regularised SENSE)

SparseMRI

iteration for J(x)

Calculate J(x) comp. independently ~ Summation

Variable Splitting (Linear Operator) ex) FS, R

P1 : $\min\{J_1(x, u_1)\}$ s.t $u_1 = Rx$ where $J_1(x, u_1) = f(x) + \varphi(u_1)$

$= \frac{1}{2}\|d - FSx\|_2^2 + \sum \lambda \sum \phi(\sum [u_1]^m)$

AL

$\mathcal{L}_1(x, u_1, \gamma_1, \mu) = \mathcal{L}_1(u, \gamma_1, \mu) = J_1(u) + \gamma_1^H Cu + \frac{\mu}{2}\|Cu\|_2^2$ where $u = \begin{bmatrix} u_1 \\ x \end{bmatrix}$, $C = [I_R \; -R]$, $b = 0$

$\eta_1 = -\frac{1}{\mu}\gamma_1$, Grouping 'Cu' tgt

$\mathcal{L}_1(u, \eta_1, \mu) = J_1(u) + \frac{\mu}{2}\|Cu - \eta_1\|_2^2 \Longrightarrow$

iteration

$u^{j+1} = \arg\min \mathcal{L}_1(u, \eta_1, \mu)$

$\eta^{j+1} = \eta^j - (Cu^{j+1} - b)$

$= \eta^j - (u_1^{j+1} - Rx^{j+1})$

$\xrightarrow{u = \begin{bmatrix} u_1 \\ x \end{bmatrix}}$

iteration

① w.r.t. x : l_2-norm → Analytic Sol'n

② w.r.t. u_1 : l_p-norm → Shrinkage rule

LUT

iteration for inverse matrix G^{-1} but pre-calculated

B. Splitting the Fourier Encoding and Spatial Components in the Data-Fidelity Term

$$\mathbf{P1}: \min_{\mathbf{u}_1,\mathbf{x}} J_1(\mathbf{x},\mathbf{u}_1) \text{ subject to } \mathbf{u}_1 = \mathbf{R}\mathbf{x}$$

where

$$J_1(\mathbf{x},\mathbf{u}_1) \triangleq \frac{1}{2}\|\mathbf{d}-\mathbf{F}\mathbf{S}\mathbf{x}\|^2 + \sum_{q=1}^{Q}\lambda_q \sum_{n=1}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{u}_{1pq}]_n|^{m_q}\right)$$

The **data-fidelity term** is composed of **different domains operator** (S and F)

So, let's introduce **auxiliary variables** to split these.

$$\mathbf{P2}: \min_{\mathbf{u}_0,\mathbf{u}_1,\mathbf{u}_2,\mathbf{x}} J_2(\mathbf{u}_0,\mathbf{u}_1) \text{ subject to}$$

$$\mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{R}\mathbf{u}_2 \text{ and } \mathbf{u}_2 = \mathbf{x}$$

where $\mathbf{u}_0 \in \mathbb{C}^{NL}$, $\mathbf{u}_1 \in \mathbb{C}^{R}$, $\mathbf{u}_2 \in \mathbb{C}^{N}$, and

$$J_2(\mathbf{u}_0,\mathbf{u}_1) \triangleq \frac{1}{2}\|\mathbf{d}-\mathbf{F}\mathbf{u}_0\|^2 + \sum_{q=1}^{Q}\lambda_q \sum_{n=1}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{u}_{1pq}]_n|^{m_q}\right).$$

B. Splitting the Fourier Encoding and Spatial Components in the Data-Fidelity Term

In terms of the general AL formulation,

$$\mathcal{L}_2(\mathbf{u}, \boldsymbol{\gamma}_2, \mu) = J_2(\mathbf{u}_0, \mathbf{u}_1) + \boldsymbol{\gamma}_1^{\mathrm{H}} \boldsymbol{\Lambda} \mathbf{B} \mathbf{u} + \frac{\mu}{2} ||\mathbf{B}\mathbf{u}||^2_{\boldsymbol{\Lambda}^2}$$

where

$$\mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{R}\mathbf{u}_2 \text{ and } \mathbf{u}_2 = \mathbf{x}$$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{x} \end{bmatrix}, \ f(\mathbf{u}) = J_2(\mathbf{u}_0, \mathbf{u}_1), \ \mathbf{C} = \boldsymbol{\Lambda}\mathbf{B}, \ \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sqrt{\nu_1}\mathbf{I}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2}\mathbf{I}_N \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \mathbf{I}_R & -\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix}.$$

$$\boldsymbol{\gamma}_2 = [\boldsymbol{\gamma}_{20}^{\mathrm{H}} \ \boldsymbol{\gamma}_{21}^{\mathrm{H}} \ \boldsymbol{\gamma}_{22}^{\mathrm{H}}]^{\mathrm{H}}$$

Diagonal weighting matrix **Λ**

It does not alter problem P2 as long as v>0.

v specifies the **relative influence** of the constraints individually.

On the other hand, $\boldsymbol{\mu}$ specifies the **overall influence** of the constraints.

Role of **preconditioner**, doesn’t affect to the final solution.

B. Splitting the Fourier Encoding and Spatial Components in the Data-Fidelity Term

Without irrelevant constant,

$$\mathcal{L}_2(\mathbf{u}, \boldsymbol{\eta}_2, \mu) = J_2(\mathbf{u}_0, \mathbf{u}_1) + \frac{\mu}{2}||\mathbf{B}\mathbf{u} - \boldsymbol{\eta}_2||^2_{\boldsymbol{\Lambda}^2} \qquad (25)$$

or

$$£_2(\boldsymbol{u}, \boldsymbol{\eta}_2; \mu) = J_2(\boldsymbol{u}_0, \boldsymbol{u}_1) + \frac{\mu}{2}||C\boldsymbol{u} - \tilde{\boldsymbol{\eta}}_2||_2^2$$

where

$$\tilde{\boldsymbol{\eta}}_2 = [\tilde{\boldsymbol{\eta}}_{20}^{\mathrm{H}}\ \tilde{\boldsymbol{\eta}}_{21}^{\mathrm{H}}\ \tilde{\boldsymbol{\eta}}_{22}^{\mathrm{H}}]^H = -\frac{1}{\mu}\boldsymbol{\gamma}_2$$

where

$$\mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{R}\mathbf{u}_2 \text{ and } \mathbf{u}_2 = \mathbf{x}$$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{x} \end{bmatrix}, \ f(\mathbf{u}) = J_2(\mathbf{u}_0, \mathbf{u}_1), \ \mathbf{C} = \boldsymbol{\Lambda}\mathbf{B}, \ \mathbf{b} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

$$\boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sqrt{\nu_1}\mathbf{I}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2}\mathbf{I}_N \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \mathbf{I}_R & -\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix}.$$

$$\boldsymbol{\gamma}_2 = [\boldsymbol{\gamma}_{20}^{\mathrm{H}}\ \boldsymbol{\gamma}_{21}^{\mathrm{H}}\ \boldsymbol{\gamma}_{22}^{\mathrm{H}}]^{\mathrm{H}}$$

$$\boldsymbol{\eta}_2 \triangleq [\boldsymbol{\eta}_{20}^{\mathrm{H}}\ \boldsymbol{\eta}_{21}^{\mathrm{H}}\ \boldsymbol{\eta}_{22}^{\mathrm{H}}]^{\mathrm{H}} = -(1/\mu)\boldsymbol{\Lambda}^{-1}\boldsymbol{\gamma}_2$$

B.1. Minimization w.r.t. $u_{0,2}$ and x

Algorithm AL

1. Select $\mathbf{u}^{(0)}$, $\boldsymbol{\eta}^{(0)}$, and $\mu > 0$; set $j = 0$

Repeat

2. $\mathbf{u}^{(j+1)} = \arg\min_{\mathbf{u}}\{f(\mathbf{u}) + (\mu/2)\|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}^{(j)}\|_2^2\}$
3. $\boldsymbol{\eta}^{(j+1)} = \boldsymbol{\eta}^{(j)} - (\mathbf{C}\mathbf{u}^{(j+1)} - \mathbf{b})$
4. Set $j = j + 1$

Until stop-criterion is met

Alternating minimization to (25)

$$\mathbf{u}_0^{(j+1)} = \arg\min_{\mathbf{u}_0}\left\{\begin{array}{l}\frac{1}{2}\|\mathbf{d} - \mathbf{F}\mathbf{u}_0\|_2^2 \\ +\frac{\mu}{2}\|\mathbf{u}_0 - \mathbf{S}\mathbf{x}^{(j)} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2\end{array}\right\} \tag{26}$$

$$\mathbf{u}_1^{(j+1)} = \arg\min_{\mathbf{u}_1}\left\{\begin{array}{l}\sum_{q=1}^{Q}\lambda_q\sum_{n=q}^{N_q}\Phi_{qn}\left(\sum_{p=1}^{P_q}|[\mathbf{u}_{1pq}]_n|^{m_q}\right) \\ +\frac{\mu\nu_1}{2}\|\mathbf{u}_1 - \mathbf{R}\mathbf{u}_2^{(j)} - \boldsymbol{\eta}_{21}^{(j)}\|_2^2\end{array}\right\} \tag{27}$$

$$\mathbf{u}_2^{(j+1)} = \arg\min_{\mathbf{u}_2}\left\{\begin{array}{l}\frac{\mu\nu_1}{2}\|\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{u}_2 - \boldsymbol{\eta}_{21}^{(j)}\|_2^2 \\ +\frac{\mu\nu_2}{2}\|\mathbf{u}_2 - \mathbf{x}^{(j)} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2\end{array}\right\} \tag{28}$$

$$\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}}\left\{\begin{array}{l}\frac{\mu}{2}\|\mathbf{u}_0^{(j+1)} - \mathbf{S}\mathbf{x} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2 \\ +\frac{\mu\nu_2}{2}\|\mathbf{u}_2^{(j+1)} - \mathbf{x} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2\end{array}\right\}. \tag{29}$$

The cost function in (26), (28), and (29) are all **quadratic**, therefore, **closed-form solutions** exist.

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1}\left[\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu(\mathbf{S}\mathbf{x}^{(j)} + \boldsymbol{\eta}_{20}^{(j)})\right] \tag{30}$$

$$\mathbf{u}_2^{(j+1)} = \mathbf{H}_{\nu_1\nu_2}^{-1}\left[\mathbf{R}^{\mathrm{H}}(\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_{21}^{(j)}) + \frac{\nu_2}{\nu_1}(\mathbf{x}^{(j)} + \boldsymbol{\eta}_{22}^{(j)})\right] \tag{31}$$

$$\mathbf{x}^{(j+1)} = \mathbf{H}_{\nu_2}^{-1}\left[\mathbf{S}^{\mathrm{H}}(\mathbf{u}_0^{(j+1)} - \boldsymbol{\eta}_{20}^{(j)}) + \nu_2(\mathbf{u}_2^{(j+1)} - \boldsymbol{\eta}_{22}^{(j)})\right] \tag{32}$$

where

Cf. inverse matrix of P1

$$\mathbf{G}_\mu = \mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{F}\mathbf{S} + \mu\mathbf{R}^{\mathrm{H}}\mathbf{R}. \tag{18}$$

$$\mathbf{H}_\mu = \mathbf{F}^{\mathrm{H}}\mathbf{F} + \mu\mathbf{I}_{NL} \tag{33}$$

$$\mathbf{H}_{\nu_1\nu_2} = \mathbf{R}^{\mathrm{H}}\mathbf{R} + \frac{\nu_2}{\nu_1}\mathbf{I}_N \tag{34}$$

$$\mathbf{H}_{\nu_2} = \mathbf{S}^{\mathrm{H}}\mathbf{S} + \nu_2\mathbf{I}_N. \tag{35}$$

All inverse matrices are separated into **in-domain operation.**

Proposed **v** ensure balance between the various constraints (block rows of B may be different orders of magnitude).

By adjusting $\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_2$, the condition number of H_{ν_1,ν_2} and H_{ν_2} can be better (higher stability of the inverses)

B.2. Implementing the matrix inverses

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1}\left[\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu(\mathbf{S}\mathbf{x}^{(j)} + \boldsymbol{\eta}_{20}^{(j)})\right] \qquad (30)$$

where

$$\mathbf{H}_\mu = \mathbf{F}^{\mathrm{H}}\mathbf{F} + \mu\mathbf{I}_{NL} \qquad (33)$$

For non-Cartesian k-space trajectories, computing $\boldsymbol{u}_0^{(j+1)}$ requires an iterative method like gridding based CG method.

Or, we can exploit the special structure of $\boldsymbol{F}^{\boldsymbol{H}}\boldsymbol{F}$ to implement.

$$\mathbf{F}^{\mathrm{H}}\mathbf{F} = \mathbf{Z}^{\mathrm{H}}\mathbf{Q}\mathbf{Z} \qquad (36)$$

where $\mathbf{Z}$ is a $2NL \times NL$ zero-padding matrix

$\mathbf{Q}$ is a $2NL \times 2NL$ circulant matrix

Then,

$$\mathbf{H}_\mu = \left(\mathbf{Z}^{\mathrm{H}}\mathbf{Q}_1\mathbf{Z} + \frac{\mu}{2}\mathbf{I}_{NL}\right)$$

where $\mathbf{Q}_1 = \mathbf{Q} + (\mu/2)\mathbf{I}_{2NL}$

// split the μ-factor because Q may have a nontrivial null-space (non-invertible)

Let $\boldsymbol{\omega}$ denote the brackets on the RHS of (30) and we apply Sherman-Morrison-Woodbury matrix inversion lemma(MIL)

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1}\boldsymbol{w} = \frac{2}{\mu}\boldsymbol{w} - \frac{4}{\mu^2}\mathbf{Z}^{\mathrm{H}}\boldsymbol{\tau} \qquad (37)$$

where $\boldsymbol{\tau}$ must be obtained by solving

$$\left(\mathbf{Q}_1^{-1} + \frac{2}{\mu}\mathbf{Z}\mathbf{Z}^{\mathrm{H}}\right)\boldsymbol{\tau} = \mathbf{Z}\boldsymbol{w}. \qquad (38)$$

Since $\boldsymbol{Q_1}$ is circulant and $\boldsymbol{ZZ^H}$ is a diagonal matrix containing either ones or zeros, we use a circulant preconditioner of the form $(\boldsymbol{Q_1^{-1}} + \boldsymbol{\alpha I_{2NL}})^{-1}$

The advantage here is that the matrices in the LHS of (38) and preconditioner are either circulant or diagonal, which simplifies CG-implementation

B.2. Implementing the matrix inverses

$$\textbf{P1}: \quad \mathcal{L}_1(\mathbf{u}, \boldsymbol{\eta}_1, \mu) = J_1(\mathbf{x}, \mathbf{u}_1) + \frac{\mu}{2}\|\mathbf{C}\mathbf{u} - \boldsymbol{\eta}_1\|^2 \tag{14}$$

where $\boldsymbol{\eta}_1 = -(1/\mu)\boldsymbol{\gamma}_1$

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{x} \end{bmatrix}, \; f(\mathbf{u}) = J_1(\mathbf{x}, \mathbf{u}_1), \; \mathbf{C} = [\mathbf{I}_R \; - \mathbf{R}]$$

$$\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}} \left\{ \frac{1}{2}\|\mathbf{d} - \mathbf{F}\mathbf{S}\mathbf{x}\|_2^2 + \frac{\mu}{2}\left\|\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{x} - \boldsymbol{\eta}_1^{(j)}\right\|_2^2 \right\}$$

$$\mathbf{x}^{(j+1)} = \mathbf{G}_\mu^{-1}[\mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu\mathbf{R}^{\mathrm{H}}(\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_1^{(j)})] \tag{17}$$

where

$$\mathbf{G}_\mu = \mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{F}\mathbf{S} + \mu\mathbf{R}^{\mathrm{H}}\mathbf{R}. \tag{18}$$

$$\textbf{P2}: \quad \mathcal{L}_2(\mathbf{u}, \boldsymbol{\eta}_2, \mu) = J_2(\mathbf{u}_0, \mathbf{u}_1) + \frac{\mu}{2}\|\mathbf{B}\mathbf{u} - \boldsymbol{\eta}_2\|_{\boldsymbol{\Lambda}^2}^2 \tag{25}$$

$\mathbf{u}_0 = \mathbf{S}\mathbf{x}, \mathbf{u}_1 = \mathbf{R}\mathbf{u}_2$ and $\mathbf{u}_2 = \mathbf{x}$

where

$$\mathbf{u} = \begin{bmatrix} \mathbf{u}_0 \\ \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{x} \end{bmatrix}, \; f(\mathbf{u}) = J_2(\mathbf{u}_0, \mathbf{u}_1), \; \mathbf{C} = \boldsymbol{\Lambda}\mathbf{B}, \; \mathbf{b} = \begin{bmatrix} \mathbf{0} \\ \mathbf{0} \\ \mathbf{0} \end{bmatrix}$$

$$\boldsymbol{\Lambda} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \sqrt{\nu_1}\mathbf{I}_R & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \sqrt{\nu_2}\mathbf{I}_N \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{I}_{NL} & \mathbf{0} & \mathbf{0} & -\mathbf{S} \\ \mathbf{0} & \mathbf{I}_R & -\mathbf{R} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix}.$$

$$\boldsymbol{\gamma}_2 = [\boldsymbol{\gamma}_{20}^{\mathrm{H}} \; \boldsymbol{\gamma}_{21}^{\mathrm{H}} \; \boldsymbol{\gamma}_{22}^{\mathrm{H}}]^{\mathrm{H}}$$

$$\boldsymbol{\eta}_2 \triangleq [\boldsymbol{\eta}_{20}^{\mathrm{H}} \; \boldsymbol{\eta}_{21}^{\mathrm{H}} \; \boldsymbol{\eta}_{22}^{\mathrm{H}}]^{\mathrm{H}} = -(1/\mu)\boldsymbol{\Lambda}^{-1}\boldsymbol{\gamma}_2$$

$$\mathbf{u}_0^{(j+1)} = \arg\min_{\mathbf{u}_0} \left\{ \begin{array}{l} \frac{1}{2}\|\mathbf{d} - \mathbf{F}\mathbf{u}_0\|_2^2 \\ + \frac{\mu}{2}\|\mathbf{u}_0 - \mathbf{S}\mathbf{x}^{(j)} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2 \end{array} \right\} \tag{26}$$

$$\mathbf{u}_1^{(j+1)} = \arg\min_{\mathbf{u}_1} \left\{ \begin{array}{l} \sum_{q=1}^{Q} \lambda_q \sum_{n=q}^{N_q} \Phi_{qn}\left(\sum_{p=1}^{P_q} |[\mathbf{u}_{1pq}]_n|^{m_q} \right) \\ + \frac{\mu\nu_1}{2}\|\mathbf{u}_1 - \mathbf{R}\mathbf{u}_2^{(j)} - \boldsymbol{\eta}_{21}^{(j)}\|_2^2 \end{array} \right\} \tag{27}$$

$$\mathbf{u}_2^{(j+1)} = \arg\min_{\mathbf{u}_2} \left\{ \begin{array}{l} \frac{\mu\nu_1}{2}\|\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{u}_2 - \boldsymbol{\eta}_{21}^{(j)}\|_2^2 \\ + \frac{\mu\nu_2}{2}\|\mathbf{u}_2 - \mathbf{x}^{(j)} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2 \end{array} \right\} \tag{28}$$

$$\mathbf{x}^{(j+1)} = \arg\min_{\mathbf{x}} \left\{ \begin{array}{l} \frac{\mu}{2}\|\mathbf{u}_0^{(j+1)} - \mathbf{S}\mathbf{x} - \boldsymbol{\eta}_{20}^{(j)}\|_2^2 \\ + \frac{\mu\nu_2}{2}\|\mathbf{u}_2^{(j+1)} - \mathbf{x} - \boldsymbol{\eta}_{22}^{(j)}\|_2^2 \end{array} \right\}. \tag{29}$$

$$\mathbf{u}_0^{(j+1)} = \mathbf{H}_\mu^{-1}\left[\mathbf{F}^{\mathrm{H}}\mathbf{d} + \mu(\mathbf{S}\mathbf{x}^{(j)} + \boldsymbol{\eta}_{20}^{(j)})\right] \tag{30}$$

$$\mathbf{u}_2^{(j+1)} = \mathbf{H}_{\nu_1\nu_2}^{-1}\left[\mathbf{R}^{\mathrm{H}}(\mathbf{u}_1^{(j+1)} - \boldsymbol{\eta}_{21}^{(j)}) + \frac{\nu_2}{\nu_1}(\mathbf{x}^{(j)} + \boldsymbol{\eta}_{22}^{(j)})\right] \tag{31}$$

$$\mathbf{x}^{(j+1)} = \mathbf{H}_{\nu_2}^{-1}\left[\mathbf{S}^{\mathrm{H}}(\mathbf{u}_0^{(j+1)} - \boldsymbol{\eta}_{20}^{(j)}) + \nu_2(\mathbf{u}_2^{(j+1)} - \boldsymbol{\eta}_{22}^{(j)})\right] \tag{32}$$

where

$$\mathbf{H}_\mu = \mathbf{F}^{\mathrm{H}}\mathbf{F} + \mu\mathbf{I}_{NL} \tag{33}$$

$$\mathbf{H}_{\nu_1\nu_2} = \mathbf{R}^{\mathrm{H}}\mathbf{R} + \frac{\nu_2}{\nu_1}\mathbf{I}_N \tag{34}$$

$$\mathbf{H}_{\nu_2} = \mathbf{S}^{\mathrm{H}}\mathbf{S} + \nu_2\mathbf{I}_N. \tag{35}$$

The effect of auxiliary variable, $\boldsymbol{u}_0$ and $\boldsymbol{u}_2$

Without u_0 : S-F are not detachable

Without u_2 : S-R are not detachable

B.3. AL algorithm for solving P2

AL-P2: AL Algorithm for solving problem P2

1. Select $\mathbf{x}^{(0)}$, $\mathbf{u}_2^{(0)} = \mathbf{x}^{(0)}$, $\nu_{1,2} > 0$, and $\mu > 0$
2. Precompute $\mathbf{F}^{\mathrm{H}}\mathbf{d}$; set $\boldsymbol{\eta}_{20,21,22}^{(0)} = \mathbf{0}$ and $j = 0$

Repeat:

3. Compute $\mathbf{u}_0^{(j+1)}$ from **(30)** using FFTs on **(37)**
4. Compute $\mathbf{u}_1^{(j+1)}$ using an appropriate technique as described in **Section IV-A2 to IV-A6** for problem **(27)**
5. Compute $\mathbf{u}_2^{(j+1)}$ using **(31)**
6. Compute $\mathbf{x}^{(j+1)}$ using **(32)**
7. $\boldsymbol{\eta}_{20}^{(j+1)} = \boldsymbol{\eta}_{20}^{(j)} - (\mathbf{u}_0^{(j+1)} - \mathbf{S}\mathbf{x}^{(j+1)})$
8. $\boldsymbol{\eta}_{21}^{(j+1)} = \boldsymbol{\eta}_{21}^{(j)} - (\mathbf{u}_1^{(j+1)} - \mathbf{R}\mathbf{u}_2^{(j+1)})$
9. $\boldsymbol{\eta}_{22}^{(j+1)} = \boldsymbol{\eta}_{22}^{(j)} - (\mathbf{u}_2^{(j+1)} - \mathbf{x}^{(j+1)})$
10. Set $j = j + 1$

Until stop-criterion is met

C. Choosing μ- and ν- values for the AL algorithms

They do not affect the final solution but, they can affect the convergence rate.

In **P2** case (μ,ν),

$$\mathbf{H}_{\mu} = \mathbf{F}^{\mathrm{H}}\mathbf{F} + \mu \mathbf{I}_{NL} \quad (33)$$
$$\mathbf{H}_{\nu_1 \nu_2} = \mathbf{R}^{\mathrm{H}}\mathbf{R} + \frac{\nu_2}{\nu_1}\mathbf{I}_{N} \quad (34)$$
$$\mathbf{H}_{\nu_2} = \mathbf{S}^{\mathrm{H}}\mathbf{S} + \nu_2 \mathbf{I}_{N}. \quad (35)$$

With a **large scalar** in front of identity function, each matrix becomes **over-regularized** and induces **slow convergency.** $\kappa(\mathbf{H}_{\mu}) \rightarrow 1$

With too **small μ**, $\boldsymbol{H_{\mu}}$ **can be inversed** because of a nontrivial null-space of $F^H F$

In **P1** case (μ,ν),

$$\mathbf{G}_{\mu} = \mathbf{S}^{\mathrm{H}}\mathbf{F}^{\mathrm{H}}\mathbf{F}\mathbf{S} + \mu \mathbf{R}^{\mathrm{H}}\mathbf{R}. \quad (18)$$

$\boldsymbol{S^H F^H F S}$ and $\boldsymbol{R^H R}$ **balance each other** in preventing G_{μ} from having a nontrivial null-space.

—the condition number $\kappa(\mathbf{G}_{\mu})$ of $\mathbf{G}_{\mu}$ therefore exhibits a minimum for some $\mu_{\min} > 0$: $\mu_{\min} = \arg\min_{\mu} \kappa(\mathbf{G}_{\mu})$. It was suggested in [15] that $\mu_{\min}$ can be used for split-Bregman-like schemes such as **AL-P1** for ensuring quick convergence of the CG algorithm applied to (17) (Step 4 of **AL-P1**).

However, selecting $\mu = \mu_{min}$ did not consistently yield fast convergence experimentally.