An Efficient Algorithm for Compressive Sensing Based SAR Tomography

Yilei Shi, Technische Universität München (TUM-LMF), yilei.shi@tum.de, Germany
Xiao Xiang Zhu, German Aerospace Center (DLR) & Technische Universität München (TUM-SIPEO), Germany
Waotao Yin, University of Califonia, Los Angeles (UCLA-MATH), USA
Richard Bamler, German Aerospace Center (DLR) & Technische Universität München (TUM-LMF), Germany

Abstract

Modern Tomographic SAR is an advanced InSAR techniques for urban mapping, which can not only retrieve 3D spatial information but also assess the 4D temporal information, such as deformation. To retrieve the information from InSAR data, several algorithms have been developed. Among them, SL1MMER algorithms is state of the art. However, it suffers from the computational expenses and it is hard to extend to large scale practice. In this work, we propose a novel optimization algorithms for L_1 regularized least square in SL1MMER, which can keep the accuracy of the optimization result and dramatically speed up the processing.

1 Introduction

With multiple acquisitions taken from slightly different positions, advanced InSAR method Tomographic SAR (TomoSAR), allow us to retrieve information of individual scattering objects fully in 3D. Its extension, namely Differential Tomographic SAR (D-TomoSAR) can even assess deformation on the order of millimeters. This provides us for the first time the opportunity to map the cities in 3D or even 4D. Motivated by these need, several algorithms have been developed in the last decade [1]-[4]. Among them, the compressive sensing based SL1MMER algorithm stands for the state of the art, thanks to its super-resolution power and estimation accuracy. Using meter resolution SAR data, its resulting 4D point clouds have a point (scatterer) density that is comparable to LiDAR. E.g. experiments using TerraSAR-X high-resolution spotlight data stacks show that the scatterer density retrieved using TomoSAR with SL1MMER is on the order of 1 million pts/km2 [5]. Object reconstruction from these high quality TomoSAR point clouds can greatly support the reconstruction of dynamic city models that could potentially be used to monitor and visualize the dynamics of urban infrastructure in very high level of details [6].

However, TomoSAR with SL1MMER is computationally extremely expensive. In [7], Wang etc. proposed an efficient approach for TomoSAR processing, which use persistent scatterer interferometry (PSI) and support vector machine (SVM) techniques to obtain a prior knowledge of the estimates, then SVD-Wiener and SL1MMER have been applied for different pixels. Although the approach speeds up the processing, but the most heavy computational part of the processing is the sparse optimization procedure. In this work, we propose a novel approach for the sparse optimization in SL1MMER, which can keep the accuracy and super-resolution power, namely the scatterer density and speed up the whole processing to an operational level for large scale problem.

2 Methodology

2.1 Review of SL1MMER

In [4], Zhu et al. proposed "Scale-down by L_1 norm Minimization, Model selection, and Estimation Reconstruction" (SL1MMER algorithms, pronounced "slimmer"). They demonstrated its super-resolution power and robustness for spaceborne tomographic SAR in [8][9][10]. The SL1MMER algorithm improves the CS algorithm and estimates these parameters in a very accurate and robust way. It consists of three main steps: 1) a L1LS minimization; 2) model selection; and 3) parameter estimation. Among all the steps, L1LS minimization is the most time consuming one. The D-TomoSAR system model can be expressed as follows:

$$g_n = \int_{\Delta s} \gamma(s) \exp(j2\pi(\xi_n s + 2d(s,t_n)/\lambda))ds \quad (1)$$

where g_n is the complex-valued measurement. $\gamma(s)$ represents the reflectivity function. $\xi_n = 2b_n/(\lambda r)$ is the spatial frequency proportional to the respective baseline b_n and λ is the wavelength and r is the range. $d(s,t_n)$ is the line-of-sight (LOS) motion as a function of elevation and time. In the presence of noise ε, the discrete-TomoSAR system model can be rewritten as:

$$\mathbf{g} = \mathbf{R}\gamma + \varepsilon \quad (2)$$

In case there is no prior knowledge about number of scatters and in presence of measurement noise, L1LS minimization can be approximated by

$$\hat{\gamma} = \arg\min_{\gamma}\{\|\mathbf{R}\gamma - \mathbf{g}\|_{\mathbf{2}}^{\mathbf{2}} + \lambda\|\gamma\|_{\mathbf{1}}\} \quad (3)$$

Generic methods for non-differentiable convex problems, such as the ellipsoid method or subgradient methods [11][12] can be used to solve Eq. (3). These methods

are often very slow. The equation (3) can be transformed to a convex quadratic problem, with linear inequality constraints. The equivalent quadratic program (QP) can be solved by standard convex optimization methods such as interior-point methods. However, the data of InSAR is complex-valued, which requires the use of second order cone program (SOCP) instead of QP for solving Eq. (3). In [4], the second order method, namely PDIPM with self-dual embedding techniques was adopted to solve SOCP. This leads to computational expenses and difficultly extend to very large scale. To make TomoSAR processing fit for high throughput or operational use, we need a fast L1LS solver.

2.2 Randomized Blockwise Proximal Gradient Algorithms

In this section, we propose a novel approach for solving L1LS minimization, which can keep the super-resolution power of the standard BPDN solver and extremely speed up the processing for matrix $\mathbf{A}$ of the random Fourier transform as used in TomoSAR. Our unconstrained optimization problems with an objective function can be split into the convex differentiable part and the convex non-differentiable part, lead to the so-called Proximal Gradient (PG) method. The PG method is used for optimization of an unconstrained problem with an objective function $F(\mathbf{x})$ split in two components. We consider the following problem

$$\min_{\mathbf{x}} F(\mathbf{x}) = f(\mathbf{x}) + r(\mathbf{x}) \tag{4}$$

where $f(\mathbf{x})$ is convex differentiable function and $r(\mathbf{x})$ is convex and non-differentiable regularization function. The iterative approach to solve (4) can be written as

$$\mathbf{x}^{k+1} = \arg\min \left(\langle \nabla f(\mathbf{x}^k), \mathbf{x} - \mathbf{x}^k \rangle + \frac{1}{2\alpha_k} \|\mathbf{x} - \mathbf{x}^k\|_2^2 + r(\mathbf{x}) \right) \tag{5}$$

where ∇f is the partial gradient of function f. The proximal gradient formulation is

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\alpha_k r}(\mathbf{x}^k - \alpha_k \nabla f(\mathbf{x}^k)) \tag{6}$$

where $\alpha_k > 0$ is step size, can be constant or determined by line search. For $r(\mathbf{x}) = \|\mathbf{x}\|_1$, the proximal operator can be chosen as soft-thresholding

$$\mathrm{prox}_{\alpha_k r}(x) = \begin{cases} x - \alpha_k, & x > \alpha_k \\ 0, & -\alpha_k \le x \le \alpha_k \\ x + \alpha_k, & x < \alpha_k \end{cases} \tag{7}$$

Proximal gradient algorithms can be accelerated by using Nesterov's Method [13] in following way.

$$\mathbf{y}^{k+1} = \mathbf{x}^k + \theta_k \left(\frac{1}{\theta_{k-1}} - 1 \right) (\mathbf{x}^k - \mathbf{x}^{k-1}) \tag{8}$$

$$\mathbf{x}^{k+1} = \mathrm{prox}_{\alpha_k r}(\mathbf{y}^{k+1} - \alpha_k \nabla f(\mathbf{y}^{k+1})) \tag{9}$$

where θ_k is chosen as $2/(k+1)$. The convergence rate of the basic PG algorithms is improved to $O(1/k^2)$ by the extrapolation. In order to further accelerate the algorithms, randomized block coordinate is adopted. As shown in [14][15], by applying block coordinate techniques, the equation (5) can be written as

$$\begin{aligned} \mathbf{x}_{i_k}^{k+1} = \quad & \arg\min \Big(\langle \nabla f_{i_k}(\mathbf{x}_{i_k}^k), \mathbf{x}_{i_k} - \mathbf{x}_{i_k}^k \rangle \\ & + \tfrac{1}{2\alpha_{i_k}^k} \|\mathbf{x}_{i_k} - \mathbf{x}_{i_k}^k\|_2^2 + r_{i_k}(\mathbf{x}) \Big) \end{aligned} \tag{10}$$

where i_k is the index of block. The choice of the update index i_k for each iteration is crucial for good performance. Often, it is easy to switch index orders. However, the choice of index affects convergence, possibly resulting in faster convergence or divergence. In this work, we choose randomized variants scheme, which has the strengths, such as less memory consumption, good convergence performance and empirically avoids local optimal. i_k is chosen randomly following the probability distribution given by the vector

$$P_{i_k} = \frac{L_{i_k}}{\sum_{j=1}^{J} L_j}, \quad i_k = 1, ..., J \tag{11}$$

where L_{i_k} is the Lipschitz constant of $\nabla_{i_k} f(\mathbf{x})$, the gradient of $f(\mathbf{x})$ with respect to the i_k-th group (in our case $L = ||A^T A||$). However, setting $\alpha_k = 1/L$ usually results in very small step sizes; Hence, the time step α_k is adaptively chosen by using backtracking line search method in [16].
The step length α_k is determined so that the following holds

$$f(x') \le f(x) + \nabla f(x)^T (x' - x) + \frac{1}{2\alpha_k} ||x' - x||^2 \tag{12}$$

This condition ensures that the value $f(x')$ of f at the new point x' is smaller than the value of quadratic approximation at the point x. The framework of our method is given in Algorithm 1.

Algorithm 1 RBPG with backtracking

Init: $x^{(0)}, y^{(0)} = 0$; for $k \ge 1$, repeat the steps
for $k = 1, 2, ..., N_K$ **do**

$i_k \leftarrow P_{i_k} = \frac{L_{i_k}}{\sum_{j=1}^{J} L_j}$
$\mathbf{y}_{i_k}^{k+1} \leftarrow \mathbf{x}_{i_k}^k + \theta_k (\frac{1}{\theta_{k-1}} - 1)(\mathbf{x}_{i_k}^k - \mathbf{x}_{i_k}^{k-1})$
$\bar{\mathbf{x}}_{i_k}^{k+1} \leftarrow \mathrm{prox}_{\alpha_k r}(\mathbf{y}_{i_k}^{k+1} - \alpha_k \nabla f(\mathbf{y}_{i_k}^{k+1}))$

while (Eq. (12) is fulfilled) **do**
$\bar{\alpha}_k = C_\alpha \cdot \alpha_k$
repeat steps $4, 5$
end while

if $(F(\bar{\mathbf{x}}_{i_k}^{k+1}) \le F(\mathbf{x}_{i_k}^k))$ **then**
$\mathbf{x}_{i_k}^{k+1} = \bar{\mathbf{x}}_{i_k}^{k+1}$
else
$\mathbf{x}_{i_k}^{k+1} = \mathbf{x}_{i_k}^k$
end if

end for

3 Experiments

3.1 Simulation

In this section, we compare RBPG approach to SOCP and SVD approach using simulated data. The inherent (Rayleigh) elevation resolution ρ_s of the tomographic arrangement is related to the elevation aperture extent Δb [8]

$$\rho_s = \frac{\lambda r}{\Delta b} \tag{13}$$

The normalized distance is defined as

$$\kappa = \frac{s}{\rho_s} \tag{14}$$

we assume the situation with two scatterers inside an azimuth-range pixel: one scatterer located at the building facade and another from the ground with three different normalized distance $\kappa = 1.2, 0.8, 0.4$ and a number of acquisitions $N = 29$. Fig. 1 shows the comparison of the reconstructed reflectivity profiles along the elevation direction by SVD (blue solid lines), RBPG (green dash lines) and SOCP (red solid lines).

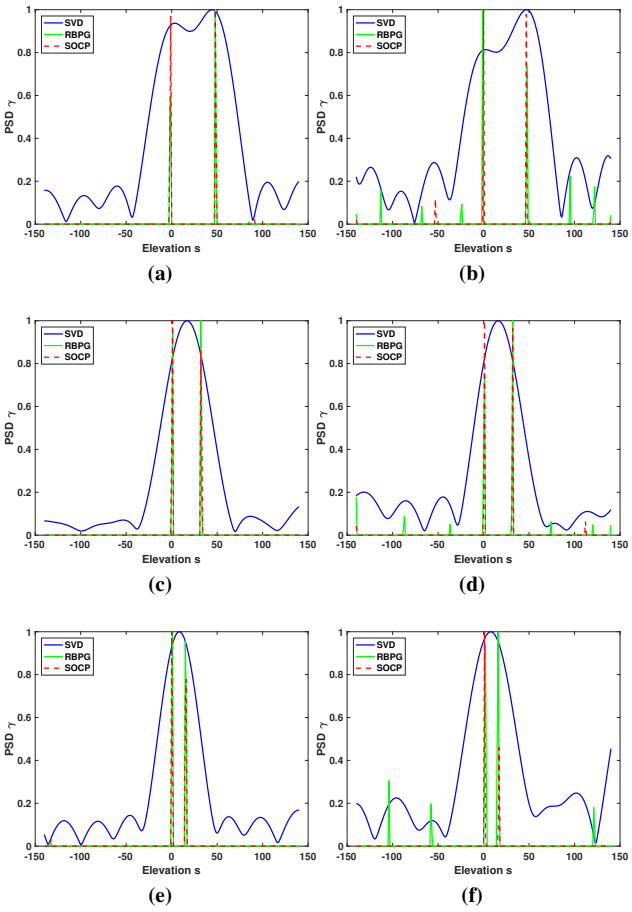

Figure 1: Performance comparison between SVD (blue), RBGP (green) and SOCP (red) on simulated data with two scatterers. (a)(c)(e) with SNR = 10 dB, (b)(d)(f) with SNR = 3 dB, the normalized distance $\kappa = 1.2, 0.8, 0.4$ from top to bottom.

From Fig. 1 one can see that all methods can distinguish the two scatterers well when $\kappa = 1.2$. However, once they move close into one elevation resolution cell, SVD failed to detect double scatterers when $\kappa = 0.8, 0.4$ for both low and high signal to noise ratio (SNR) conditions. In contrast, SOCP and RBGP can accurately estimate the position of double scatterers for all the cases, which exhibits the super resolution power.

3.2 Real Data

For the real data experiment, we choose a large scale test area covering the Munich city. The stack is composed of 78 images and covers approximately 50 km^2. Four dimensional point clouds with a density of more than two million points per square kilometer are reconstructed. The experiments carried out on the high performance computer at Lebnitz-Rechnung-Zentrum (LRZ) with thousand cores. With the same number of cores, the run time by SOCP approach is more than 120 CPU hours, whereas RBPG took only 6 CPU hours. The speed up of the new approach is 20 for the large scale case.

4 Conclusions

In this work, as a crucial step towards large scale 4D urban mapping using SAR tomography, we proposed a novel optimization approach to speed up the SL1MMER algorithm in TomoSAR processing. Experiments with simulated data and real data demonstrate that the new approach retains the super-resolution power of SL1MMER and speeds it up for 10 to 100 times, which allows to an operational level processing for large scale problem. Combing the proposed optimization approach with the processing strategy proposed in [7], a further speed-up of about 50 times can be expected. Furthermore, the proposed algorithm can be generally used for spectral estimation for other applications.

5 Acknowledgment

This work is supported by the Bavaria California Technology Center (Project: Large-Scale Problems in Earth observation), the European Research Council (ERC) under the European Union's Horizon 2020 research and innovation programme (grant agreement No. ERC-2016-StG-714087, Acronym: So2Sat), and Helmholtz Association under the framework of the Young Investigators Group "SiPEO" (VHNG-1018, www.sipeo.bgu.tum.de). The authors thank the Gauss Centre for Supercomputing (GCS) e.V. for funding this project by providing computing time on the GCS Supercomputer SuperMUC at the Leibniz Supercomputing Centre (LRZ).

References

[1] G. Fornaro, F. Lombardini, and F. Serafino: "Three-dimensional multipass SAR focusing: experiments

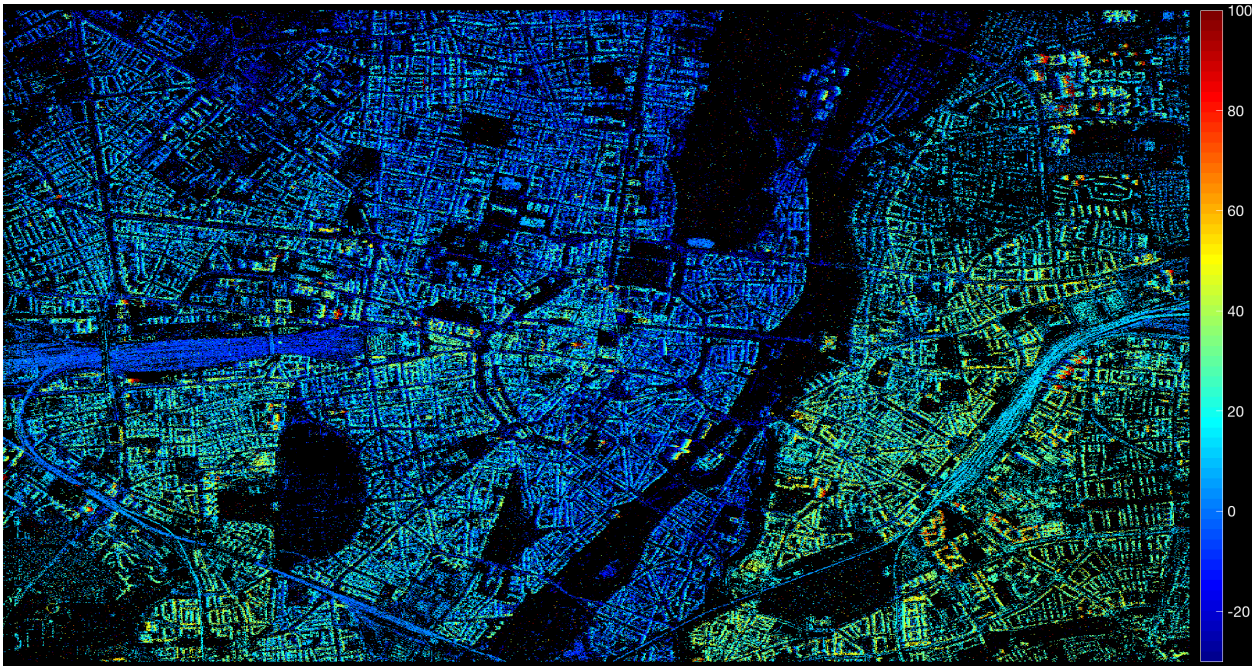

Figure 2: Test Area: Munich. reconstructed point cloud with height color-coded

with long-term spaceborne data", IEEE Trans. Geosci. Remote Sens., 43(4), p. 702-714, 2005.

[2] A. Budillon, A. Evangelista, and G. Schirinzi: "Three-dimensional SAR focusing from multipass signals using compressive sampling", IEEE Trans. Geosci. Remote Sens., 49(1), p. 488-499, 2011.

[3] X. X. Zhu and R. Bamler: "Very high resolution spaceborne SAR tomography in urban environment", IEEE Trans. Geosci. Remote Sens., 48(12), p. 4296-4308, 2010.

[4] X. X. Zhu and R. Bamler: "Tomographic SAR inversion by L1 norm regularization-The compressive sensing approach", IEEE Trans. Geosci. Remote Sens., 48(10), p. 3839-3846, 2010.

[5] X. X. Zhu, M. Shahzad: "Facade Reconstruction Using Multiview Spaceborne TomoSAR Point Clouds", IEEE Trans. Geosci. Remote Sens., 52(6), p. 3541-3552, 2014.

[6] M. Shahzad, X. X. Zhu: "Automatic Detection and Reconstruction of 2D/3D Building Shapes from Spaceborne TomoSAR Point Clouds", IEEE Trans. Geosci. Remote Sens., 54(3), p. 1292-1310, 2016.

[7] Y. Wang, X. X. Zhu, R. Bamler: "An Efficient Tomographic Inversion Approach for Urban Mapping Using Meter Resolution SAR Image Stacks", IEEE Geosci. Remote Sensing Lett., 11(7), p. 1250-1254, 2014.

[8] X. X. Zhu and R. Bamler: "Super-resolution power and robustness of compressive sensing for spectral estimation with application to spaceborne tomographic SAR", IEEE Trans. Geosci. Remote Sens., 50(1), p. 247-258, 2012.

[9] X. X. Zhu and R. Bamler: "Demonstration of superresolution for tomographic SAR imaging in urban environment", IEEE Trans. Geosci. Remote Sens., 50(8), p. 3150-3157, 2012.

[10] X. X. Zhu and R. Bamler: "Superresolving SAR tomography for multidimensional imaging of urban areas: Compressive sensing-based tomoSAR inversion", IEEE Signal Process. Mag., 31(4), p. 51-58, 2014.

[11] N. Z. Shor: "Minimization Methods for Non-Differentiable Functions", Book, Springer Series in Computational Mathematics, vol 3, 1985.

[12] B. Polyak: "Introduction to Optimization. Optimization Software", Book, Optimization Software, Inc. 1987.

[13] Y. Nesterov: "A method of solving a convex programming problem with convergence rate $O(1/k^2)$", In Soviet Mathematics Doklady, vol. 27, pp 372-376, 1983.

[14] Z. Peng, T. Wu, Y. Xu, M. Yan, and W. Yin: "Coordinate Friendly Structures, Algorithms and Applications", Annals of Mathematical Sciences and Applications, vol. 1, pp 57-119, 2016.

[15] Y. Xu and W. Yin: "Block stochastic gradient iteration for convex and nonconvex optimization", SIAM Journal on Optimization, vol. 25, pp 1686-1716, 2015.

[16] Y. Xu and W. Yin: "A globally convergent algorithm for nonconvex optimization based on block coordinate update", Journal of Scientific Computing, pp 1-35, 2017.