A SPLITTING METHOD FOR ORTHOGONALITY CONSTRAINED PROBLEMS

RONGJIE LAI* AND STANLEY OSHER†

Abstract. Orthogonality constrained problems are widely used in science and engineering. However, it is challenging to solve these problems efficiently due to the non-convex constraints. In this paper, a splitting method based on Bregman iteration is represented to tackle the optimization problems with orthogonality constraints. With the proposed method, the constrained problems can be iteratively solved by computing the corresponding unconstrained problems and orthogonality constrained quadratic problems with analytic solutions. As applications, we demonstrate the robustness of our method in several problems including direction fields correction, noisy color image restoration and global conformal mapping for genus-0 surfaces construction. Numerical comparisons with existing methods are also conducted to illustrate the efficiency of our algorithms.

Key words. Spherical constraint, orthogonality constraint, Bregman iteration, conformal mapping, $L1$-harmonic energy minimization

1. Introduction. Orthogonality constrained optimizations are widely considered in many problems such as conformal geometry [1, 2, 3], p-harmonic flow [4, 5, 6, 7, 8, 9, 10, 11, 12, 13], 1-bit compressive sensing [14, 15, 16], linear and nonlinear eigenvalue problem [17, 18], as well as combinatorial optimization [19, 20]. Mathematically, the orthogonality constrained problems can be formulated as the spherical constrained version:

$$\min_{X=(x_1,\cdots,x_m)\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad \|x_i\|_2^2 = 1, \quad i = 1,\cdots,m \tag{1.1}$$

or the matrix orthogonality constrained version:

$$\min_{X\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad X^TQX = I \tag{1.2}$$

where $\mathcal{J}$ is convex, $\|x_i\|_2$ is the l_2 norm of the $i-$th column vector of X, Q is a symmetric positive definite matrix and I is the $m \times m$ identity matrix.

It is challenging to solve the orthogonality constrained problems (1.1) and (1.2) efficiently due to the nonlinear and nonconvex constraints, which may lead to many different local minimizers as solutions. Iterative approaches are commonly used to solve optimization problems. However, it is not straightforward to generate a sequence of points preserving the nonlinear constraints and decreasing the cost functional $\mathcal{J}$. To avoid directly handling the nonlinear constraints, various methods are introduced to tackle the orthogonality constrained problems by solving a sequence of unconstrained problems to approach the feasible condition. Penalty methods [21, 22] are proposed to approximate the optimization problems (1.1) and (1.2) by unconstrained problems (1.3) and (1.4) respectively, as $\epsilon \to 0$.

$$\min_{X=(x_1,\cdots,x_m)\in\mathbb{R}^{n\times m}} \mathcal{J}(X) + \frac{1}{\epsilon}\sum_{i=1}^{m}(\|x_i\|_2^2 - 1)^2 \tag{1.3}$$

$$\min_{X\in\mathbb{R}^{n\times m}} \mathcal{J}(X) + \frac{1}{\epsilon}\|X^TQX - I\|_F^2 \tag{1.4}$$

However, penalty methods usually suffer from slow convergence since it has to solve a sequence of problems

*Department of Mathematics, University of Southern California, Los Angeles, USA. (rongjiel@usc.edu).
†Department of Mathematics, University of California, Los Angeles, USA. (sjo@math.ucla.edu).

as ϵ goes to zero. Another approach comes from the standard augmented Lagrangian method [23, 24], which tackles the constrained problems (1.1), (1.2) by solving the following unconstrained problems (1.5), (1.6) and iteratively updating the Lagrangian multipliers, respectively.

$$\min_{X=(x_1,\cdots,x_m)\in\mathbb{R}^{n\times m}} \mathcal{J}(X) + \frac{r}{2}\sum_{i=1}^{m}(\|x_i\|_2^2-1)^2 + \sum_{i=1}^{m}\lambda_i(\|x_i\|_2^2-1); \qquad \lambda_i \leftarrow \lambda_i + r(\|x_i\|_2^2-1) \quad (1.5)$$

$$\min_{X\in\mathbb{R}^{n\times m}} \mathcal{J}(X) + \frac{r}{2}\|X^TQX-I\|_F^2 + Tr(\Lambda^T(X^TQX-I)); \qquad \Lambda \leftarrow \Lambda + r(X^TQX-I) \quad (1.6)$$

Because of the complicated form of the augmented Lagrangian terms, the subproblems usually cannot be solved analytically. Additional inner iterations need to be introduced to solve subproblems, which slow down the computation. More recently, constraint preserving algorithms are proposed based on the study of the Stiefel manifold structures of the orthogonality constraint [17, 25, 26, 27]. To the best of our knowledge, the state-of-the-art method of the constraint preserving algorithms is proposed in [13, 28], where a curvilinear search approach is introduced based on the Cayley transformation combined with chosen of Barzilai-Borwein step size. The sophisticated techniques in feasible approaches are mathematically elegant but depend on the manifold structure of the Stiefel manifold. It is not straightforward to have similar algorithms if additional constraints are imposed.

In this paper, we introduce a new approach, which is called the method of splitting orthogonality constraints (SOC), to solve the orthogonality constrained problems. The idea is motivated from our recent paper [29], where an efficient algorithm is proposed to solve moving interface problems using the level set method with constraint $\|\nabla\phi\|_2 = 1$ for a distance preserving property. This idea can also be traced back to the earlier work of Glowinksi et. al. in solving variational problems from nonlinear elasticity [30, 24]. We further observe that the orthogonality constrained problems can be iteratively approximated by non-constrained problems and quadratic problems with the orthogonality constraints using Bregman iteration [31, 32]. In many applications, the obtained non-constrained problems can be solved efficiently. More importantly, the quadratic problems with orthogonality constraints can be solved analytically. This SOC method overcomes the limitations of penalty methods and the standard augmented Lagrangian methods. It also successfully avoids handling the complicated manifold structure of the constraints as the feasible approaches need to do. Although we currently have no proof of convergence for this method, our numerical experiments strongly validate its value and provide evidence of its correctness. This will motivate future theoretical analysis. According to our experience, the SOC method has the following advantages. First, the orthogonality constraints can be analytically solved using the SOC method, thus the algorithm is simple and easy to code compared with the sophisticated optimization techniques introduced in [28]. Second, the idea of SOC can be easily adapted to other problems as long as the corresponding quadratic problem can be efficiently solved. Last but not least, given the error forgetting of Bregman iteration for l_1 related problems [33], our SOC algorithms also benefit from this nice property and can efficiently solve l_1 related problems according to our numerical experiments.

The rest of this paper is organized as follows. In section 2, after a brief review of Bregman iteration, we first introduce the idea of the SOC method for spherical constrained problems. We then generalize it to matrix orthogonality constrained problems and spherical and linear equality constrained problems. In section 3, the SOC method is applied to design algorithms for three applications, finding global conformal mapping, correcting direction fields and restoring noisy color images. We also conduct numerical comparison with existing methods to demonstrate the efficiency of the proposed algorithms. Finally, conclusions and

future work are discussed in section 4.

2. The Method of Splitting Orthogonality Constraints (SOC). In this section, we propose the method of splitting orthogonality constraints (SOC) to tackle optimization problems with orthogonality constraints. Using the combination of variable splitting and Bregman iteration [31, 32], the SOC method solves the orthogonality constrained problems by iteratively optimizing unconstrained problems and quadratic problems with analytic solutions. To make this paper self-contained, we first would like to give a brief review of Bregman iteration. After that, we will introduce the idea of SOC for spherical constrained problems, then generalize it to matrix orthogonality constrained problems. In addition, we will also demonstrate the adaptability of the SOC idea to problems with spherical and linear equality constraints.

2.1. Background of Bregman iteration. Bregman iteration was first introduced into information science by S. Osher et al. in [31] for solving total variation related problems in image processing. It has attained intensive attention due to its efficiency in many l_1 related constrained optimization problems which can be typically written as follows [32, 33] :

$$\arg\min_x \mathcal{J}(x), \quad s.t. \quad \mathcal{D}x = f \tag{2.1}$$

with a convex functional $\mathcal{J}(x)$ and a linear operator $\mathcal{D}$. The optimizer of the above problem can be efficiently approached using the the following Bregman iteration method:

$$\begin{aligned} x^{k+1} &= \arg\min_x \mathfrak{B}_{\mathcal{J}}^{p^k}(x, x^k) + \frac{r}{2}\|\mathcal{D}x - f\|_2^2 \\ p^{k+1} &= p^k - r\mathcal{D}^T(\mathcal{D}x^{k+1} - b) \end{aligned} \tag{2.2}$$

where $\mathfrak{B}_{\mathcal{J}}^{p^k}(x, x^k) = \mathcal{J}(x) - \mathcal{J}(x^k) - \langle p^k, x - x^k\rangle$ is the Bregman distance [34]. It was shown that the above scheme is equivalent to a simple two step procedure with Bregman penalty function [32]:

$$\begin{aligned} x^{k+1} &= \arg\min_x \mathcal{J}(x) + \frac{r}{2}\|\mathcal{D}x - f + b^k\|_2^2 \\ b^{k+1} &= b^k + \mathcal{D}x^{k+1} - f \end{aligned} \tag{2.3}$$

This is an analog of "adding back the noise" in image denoising [31] and also equivalent to the well-known augmented Lagrangian method [35, 36].

2.2. Spherical constrained problems. To clearly describe the idea of SOC, we would like to first consider a simple case, an optimization problem with a single spherical constraint as follows:

$$\min_{x\in\mathbb{R}^n} \mathcal{J}(x), \quad s.t. \quad \|x\|_2 = 1 \tag{2.4}$$

where $\mathcal{J}(x)$ is convex and $\|x\|_2 = \sqrt{x^T x}$ is the l_2 norm. We observe that a typical type of quadratic problem with a spherical constraint can be solved analytically. Moreover, the original spherical constraint problem can be iteratively solved by computing a spherical constraint quadratic problem and an unconstrained problem using Bregman iteration. In other words, we introduce an auxiliary variable $p = x$ to split the orthogonality constraint, then the above problem is equivalent to:

$$\min_{x,\ p\in\mathbb{R}^n} \mathcal{J}(x), \quad s.t. \quad x = p \ \ \& \quad \|p\|_2 = 1 \tag{2.5}$$

The first constraint $x = p$ in the above problem is linear, which can be solved using Bregman iteration discussed in section 2.1. Namely, we iteratively solve:

$$(x^k, p^k) = \arg\min_{x,\ p\in\mathbb{R}^n} \mathcal{J}(x) + \frac{r}{2}\|x - p + b^{k-1}\|_2^2, \quad s.t. \quad \|p\|_2 = 1 \tag{2.6}$$

$$b^k = b^{k-1} + x^k - p^k. \tag{2.7}$$

Similar to the one-step iterative method in the alternating direction method of multipliers (ADMM) [37, 23] and split Bregman iteration [38], the problem (2.6) can be solved by iteratively minimizing with respect to x and p, which inspires us to propose the following method of splitting orthogonality constraints (SOC):

1. $x^k = \arg\min_{x\in\mathbb{R}^n} \mathcal{J}(x) + \frac{r}{2}\|x - p^{k-1} + b^{k-1}\|_2^2$.
2. $p^k = \arg\min_{p\in\mathbb{R}^n} \frac{r}{2}\|p - (x^k + b^{k-1})\|_2^2, \quad s.t. \quad \|p\|_2 = 1$.
3. $b^k = b^{k-1} + x^k - p^k$.

Here, the first problem is a convex optimization problem without constraint, which can be solved efficiently in many cases. More importantly, the second problem has analytic solution:

$$p^k = \mathcal{S}_{proj}(x^k + b^{k-1}) := \frac{x^k + b^{k-1}}{\|x^k + b^{k-1}\|_2}, \quad \text{if} \quad \|x^k - b^{k-1}\|_2 \neq 0 \tag{2.8}$$

These two properties of the SOC make it different from penalty methods and standard augmented Lagrangian methods, where the corresponding unconstrained problems have more complicated forms and may not be able to be solved efficiently. In many applications, such as those involving l_1 related problems, the SOC method can be expected to have good performance due to the error forgetting of Bregman iteration [33]. We will see this in the section 3.1.

Moreover, the SOC method can be easily adapted to solve the following problem with multiple spherical constraints:

$$\min_{X\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad \|X(:,i)\|_2 = 1, \quad i = 1, \cdots, m \tag{2.9}$$

where $\mathcal{J}$ is convex and $X(:,i)$ denotes the $i-$th column vector of X. Similarly, we introduce an auxiliary variable $P = X$ to split the multiple spherical constraint as follows:

$$\min_{X,P\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad X = P \ \ \& \ \ \|P(:,i)\|_2 = 1, \quad i = 1, \cdots, m \tag{2.10}$$

Using Bregman iteration, we propose the following algorithm to solve the multiple spherical constrained problem (2.9)

SOC Algorithm 1. *Initialize* $B^0 = 0, X^0 = P^0$

while *"not converge"* **do**

1. $X^k = \arg\min_X \mathcal{J}(X) + \frac{r}{2}\|X - P^{k-1} + B^{k-1}\|_F^2$;
2. $P^k(:,i) = \mathcal{S}_{proj}(X^k(:,i) + B^{k-1}(:,i)) = \frac{X^k(:,i) + B^{k-1}(:,i)}{\|X^k(:,i) + B^{k-1}(:,i)\|_2}; \quad i = 1, \cdots, m$
3. $B^k = B^{k-1} + (X^k - P^k)$

Using the SOC method, one can solve the spherical constrained problem by iteratively minimizing an unconstrained problem and a quadratic problem with the orthogonality constraints. More importantly, the solution of the quadratic problem with the orthogonality constraints can be analytically written as the

spherical projection. This provides an efficient way to handle the orthogonality constraints in the problem (2.9). Furthermore, this inspires us to think that more general types of orthogonality constrained problems can also be solved using the SOC method as long as the corresponding constrained quadratic problems have closed-form solutions. In the next two subsections, we choose two other typical types of orthogonality constrained problems to demonstrate the adaptability of SOC.

2.3. Matrix orthogonality constrained problems. We would like to further apply the SOC method to the following matrix orthogonality constrained problem

$$\min_{X\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad X^TQX = I \tag{2.11}$$

where $\mathcal{J}$ is convex, $n \geq m$ and $Q = L^TL$ is a symmetric positive definite matrix. This matrix orthogonality constrained problem has been widely considered in many applications such as linear and nonlinear eigenvalue problems, quadratic assignment problems and matrix valued image processing [17, 19, 20, 39]. Based on the structure of Grassmann and Stiefel manifolds, geometric methods have been discussed in [17, 25, 26, 27]. Here, we would like to consider the SOC method to this problem. By introducing $P = LX$ to split the orthogonality constraint, the above minimization is equivalent to the following problem:

$$\min_{X,P\in\mathbb{R}^{n\times m}} \mathcal{J}(X), \quad s.t. \quad LX = P \quad \& \quad P^TP = I \tag{2.12}$$

Using Bregman iteration, the above problem can be iteratively solved as follows:

1. $X^k = \arg\min_X \mathcal{J}(X) + \frac{r}{2}\|LX - P^{k-1} + B^{k-1}\|_F^2$.
2. $P^k = \arg\min_P \frac{r}{2}\|P - (LX^k + B^{k-1})\|_F^2, \quad s.t. \quad P^TP = I$;
3. $B^k = B^{k-1} + LX^k - P^k$.

where the first subproblem is a convex optimization problem without constraints and the second constrained quadratic problem has closed-form solution provided by the following theorem.

Theorem 2.1. *The constrained quadratic problem:*

$$P^* = \arg\min_{P\in\mathbb{R}^{n\times m}} \frac{1}{2}\|P - Y\|_F^2, \quad s.t. \quad P^TP = I \tag{2.13}$$

has the closed-form solution with two forms as follows:

1. $P^* = UI_{n\times m}V^T$, *where* $U \in \mathbb{R}^{n\times n}, V \in \mathbb{R}^{m\times m}$ *are two orthogonal matrices and* $D \in \mathbb{R}^{n\times m}$ *is a diagonal matrix satisfying the SVD factorization* $Y = UDV^T$.
2. *If* $rank(Y) = m$, $P^* = Y\tilde{V}\tilde{D}^{-1/2}\tilde{V}^T$, *where* $V \in \mathbb{R}^{m\times m}$ *is an orthonormal matrix and* $\tilde{D} \in \mathbb{R}^{m\times m}$ *is a diagonal matrix satisfying the SVD factorization* $Y^TY = \tilde{V}\tilde{D}\tilde{V}^T$.

[Proof:] 1. This result can also be found in [40, 25]. To make the paper self-contained, we briefly write the proof here. For Let $\hat{P} = U^TPV$. Since $\|P - Y\|_F^2 = \|U(U^TPV - D)V\|_F^2 = \|U(\hat{P} - D)V\|_F^2 = \|\hat{P} - D\|_F^2$ and $P^TP = I \iff \hat{P}^T\hat{P} = I$, it is clear that P^* in (2.13) can be given by $P^* = U\hat{P}^*V^T$, where $\hat{P}^*$ is a solution of the following problem:

$$\hat{P}^* = \arg\min_{\hat{P}\in\mathbb{R}^{n\times m}} \frac{1}{2}\|\hat{P} - D\|_F^2, \quad s.t. \quad \hat{P}^T\hat{P} = I \tag{2.14}$$

It is easy to see that the problem (2.14) has a unique solution $\hat{P}^* = I_{n\times m}$ if rank(D) = m. In the case that $rank(D) = k < m$, without loss of generality, we can assume $d_1, \cdots, d_k > 0$ and $d_{k+1}, \cdots, d_m = 0$. Then

for any orthogonal matrix $O_1 \in \mathbb{R}^{(m-k)\times(m-k)}$, $\hat{P}^* = \text{diag}\{sign(d_1), \cdots, sign(d_k), O_1\}$ is a solution of the problem (2.14). Thus, $P^* = UI_{n\times m}V^T$ is a solution of the constrained problem (2.13).

2. Consider the Lagrangian of the constrained problem (2.13): $\mathcal{L}(P,\Theta) = \frac{1}{2}\|P-Y\|_F^2 + Tr(\Theta(P^TP - Id))$, we have:

$$\begin{cases} \dfrac{\partial \mathcal{L}}{\partial P} = (P-Y) + P(\Theta+\Theta^T) = 0 \\ P^TP = I \end{cases} \quad \Longrightarrow \quad \begin{cases} P(I+\Theta+\Theta^T) = Y \\ P^TP = I \end{cases} \tag{2.15}$$

This implies: $(I+\Theta+\Theta^T)(I+\Theta+\Theta^T) = Y^TY$ and $P = Y(I+\Theta+\Theta^T)^{-1}$.

Since Y^TY is symmetric and semi-positive definite, there is a diagonal matrix $\tilde{D}$ and an orthonormal $m \times m$ matrix V with $\tilde{V}^T\tilde{V} = I$ such that $Y^TY = \tilde{V}\tilde{D}\tilde{V}^T$, then $I+\Theta+\Theta^T = \pm\tilde{V}\tilde{D}^{1/2}\tilde{V}^T$ are two square roots of Y^TY. The principle square root $I+\Theta+\Theta^T = \tilde{V}\tilde{D}^{1/2}\tilde{V}^T$ is the one we desired. If $rank(Y) = m$, then $\tilde{V}\tilde{D}^{1/2}\tilde{V}^T$ is invertible. Thus, $P = Y(\tilde{V}\tilde{D}^{1/2}\tilde{V}^T)^{-1} = Y\tilde{V}\tilde{D}^{-1/2}\tilde{V}^T$. □

According the above theorem, we propose the following algorithm to solve the problem (2.11):

SOC Algorithm 2. *Initialize* $X^0, P^0 = LX^0, B^0 = 0$

while *"not converge"* **do**

1. $X^k = \arg\min_X \mathcal{J}(X) + \frac{r}{2}\|LX - P^{k-1} + B^{k-1}\|_F^2$.
2. *Let* $Y^k = LX^k + B^{k-1}$. *Compute SVD factorization* $Y^k = UDV^T$.
3. $P^k = UI_{n\times m}V^T$
4. $B^k = B^{k-1} + LX^k - P^k$

Remark 1. *For problems with $m \ll n$, the solution of (2.13) given by the second form is more efficient than the first form.*

2.4. Spherical and linear equality constrained problems. According to the discussion in the previous two subsections, the SOC method splits the orthogonality constraint into a quadratic problem with analytic solution. This method successfully avoids studying the manifold structure of the constraint. As long as the corresponding constrained quadratic problem can be solved efficiently, the SOC method can be adapted to tackle problems with orthogonality constraints and other constraints, which the feasible approaches may not be able to handle. To demonstrate the potential application of the SOC method, we consider a spherical and linear equality constrained problem:

$$\min_{x\in\mathbb{R}^n} \mathcal{J}(x), \quad s.t. \quad Ax = f \quad \& \quad \|x\|_2 = 1 \tag{2.16}$$

where $\mathcal{J}(x)$ is convex and $A \in \mathbb{R}^{m\times n}$ is a matrix. Again, we introduce an auxiliary variable p to split constraints $Ax = b$ and $\|x\|_2 = 1$, then the above problem is equivalent to:

$$\min_{x\in\mathbb{R}^n} \mathcal{J}(x), \quad s.t. \quad x = p, \quad Ap = f \quad \& \quad \|p\|_2 = 1 \tag{2.17}$$

Similar as our previous method, we propose to solve the above problem using Bregman iteration:

1. $x^k = \arg\min_x \mathcal{J}(x) + \frac{r}{2}\|x - p^{k-1} + b^{k-1}\|_2^2$;
2. $p^k = \arg\min_p \frac{r}{2}\|p - (x^k + b^{k-1})\|_2^2 \quad s.t. \quad Ap = f \ \& \ \|p\|_2 = 1$;
3. $b^k = b^{k-1} + x^k - p^k$.

where the first problem is a convex minimization without constraints and, more importantly, the second constrained quadratic problem can have closed-form solution, which is provided in the following theorem.

Theorem 2.2. *Let A be a $m \times n$ matrix with rank k. Write a QR decomposition of A^T as*

$$A^T = \begin{bmatrix} Q_1 , & Q_2 \\ {\scriptstyle n\times k} & {\scriptstyle n\times(n-k)} \end{bmatrix} \begin{bmatrix} R_1 \\ {\scriptstyle k\times m} \\ 0 \\ {\scriptstyle (n-k)\times m} \end{bmatrix} \tag{2.18}$$

where $Q = [Q_1\ Q_2] \in \mathbb{R}^{n\times n}$ is an orthogonal matrix. Then the constrained quadratic problem:

$$p^* = \arg\min_{p\in\mathbb{R}^n} \frac{r}{2}\|p-y\|_2^2, \quad s.t. \quad Ap = f\ \&\ \|p\|_2 = 1. \tag{2.19}$$

has the following closed-form solution:

$$p^* = \sqrt{1-\|p^\#\|_2^2}\ \mathcal{S}_{proj}(\hat{p}) + p^\# \tag{2.20}$$

where $p^\# = \arg\min_{Ap=f} \|p\|_2$ and $\hat{p} = Q_2Q_2^Ty$.

[Proof]: Based on the QR decomposition of $A^T = [Q_1\ Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$ in (2.18), the column vectors of Q_2 form an orthonormal basis of the null space $\mathcal{N}_A$ of the matrix A. Therefore, $Q_2Q_2^T : \mathbb{R}^n \to \mathcal{N}_A, v \mapsto Q_2Q_2^Tv$ provides an orthogonal projection from $\mathbb{R}^N$ to $\mathcal{N}_A$. Let $q \in \mathbb{R}^n$ be a vector satisfying $Aq = f$, then $p^\# = \arg\min_{Ap=f} \|p\|_2 = p - Q_2Q_2^Tp$ which is perpendicular to $\mathcal{N}_A$. If we write $p = \tilde{p} + p^\#$, then the problem (2.19) is equivalent to the following constrained problems:

$$\begin{aligned} \tilde{p}^* &= \arg\min_{\tilde{p}\in\mathbb{R}^n} \frac{r}{2}\|\tilde{p}+p^\#-y\|_2^2, \quad s.t. \quad A(\tilde{p}+p^\#) = f \quad \& \quad \|\tilde{p}+p^\#\|_2 = 1 \\ \iff \tilde{p}^* &= \arg\min_{\tilde{p}\in\mathbb{R}^n} \frac{r}{2}\|\tilde{p}+p^\#-y\|_2^2, \quad s.t. \quad A\tilde{p} = 0 \quad \& \quad \|\tilde{p}\|_2 = \sqrt{1-\|p^\#\|_2^2} \end{aligned} \tag{2.21}$$

And $p^* = \tilde{p}^* + p^\#$. Since $\tilde{p} \in \mathcal{N}_A$, we have $\langle \tilde{p}, p^\# \rangle = 0$. Thus,

$$\begin{aligned} \tilde{p}^* &= \arg\min_{\tilde{p}\in\mathcal{N}_A} \frac{r}{2}\|\tilde{p}-y\|_2^2, \quad s.t. \quad \|\tilde{p}\|_2 = \sqrt{1-\|p^\#\|_2^2} \\ &= \sqrt{1-\|p^\#\|_2^2}\ \mathcal{S}_{proj}(\hat{p}) \end{aligned} \tag{2.22}$$

where $\hat{p}$ is the projection of y to $\mathcal{N}_A$ satisfying:

$$\hat{p} = \arg\min_{p\in\mathcal{N}_A} \frac{r}{2}\|p-y\|_2^2 = Q_2Q_2^Ty \tag{2.23}$$

Therefore, we have:

$$p^* = \sqrt{1-\|p^\#\|_2^2}\ \mathcal{S}_{proj}(\hat{p}) + p^\# \tag{2.24}$$

where $p^\# = \arg\min_{Ap=f} \|p\|_2$ and $\hat{p} = Q_2Q_2^Ty$. □

According the above theorem, we propose the following algorithm to solve the constrained problem (2.16).

SOC Algorithm 3. *Initialize $x^0 = p^0, b^0 = 0$ and compute the QR decomposition of $A^T = [Q_1\ Q_2] \begin{bmatrix} R_1 \\ 0 \end{bmatrix}$. Choose a vector $q \in \mathbb{R}^n$ satisfying $Aq = f$ and let $p^\# = q - Q_2Q_2^Tq$.*

while *"not converge"* **do**

1. $x^k = \arg\min_x \mathcal{J}(x) + \frac{r}{2}\|x - p^{k-1} + b^{k-1}\|_2^2$.
2. $p^k = \sqrt{1 - \|p^{\#}\|^2}\mathcal{S}_{proj}(\hat{p}) + p^{\#}$, *where* $\hat{p} = Q_2 Q_2^T(x^k + b^{k-1})$.
3. $b^k = b^{k-1} + x^k - p^k$.

3. Applications and Numerical Results. To demonstrate the robustness of the algorithms based on the SOC method proposed in section 2, several applications including direction field correction, color image restoration, conformal mapping construction and their numerical results will be presented in this section. In addition, comparisons with existing methods will also be conducted to show the efficiency of the algorithms. All experiments were implemented in Matlab on a PC with a 2.66GHz CPU and 4G of RAM. Here, we only illustrate applications for spherical constrained problems. Other applications involving algorithm 2 and algorithm 3 will be explored in our future work.

3.1. L1-harmonic minimization. As we mentioned before, Bregman iteration has excellent performance in l_1 related problems due to its error forgetting [33]. Thus, we can expect our SOC algorithm based on Bregman iteration will inherit this nice behavior. As a demonstration, we will show the SOC method for a typical l_1 related spherical constrained problem, $L1$-harmonic minimization.

$L1$-harmonic flow has many applications in directional diffusion, color image restoration, liquid crystal theory as well as micromagnetics [4, 5, 6, 7, 8, 9, 10, 11, 13, 12]. It can be guided by the gradient flow of the following L1-harmonic energy with orthogonality constraints:

$$\min_{\vec{F}} \mathcal{E}(\vec{F}) = \int_\Omega |\nabla \vec{F}| \mathrm{d}x, \quad s.t. \quad \|\vec{F}(x)\|^2 = 1 \tag{3.1}$$

where $\vec{F} = (f_1, \cdots, f_n) : \Omega \to \mathbb{R}^n$, $|\nabla \vec{F}(x)| = \sqrt{\sum_{i=1}^n |\nabla f_i(x)|^2}$ and $\|\vec{F}(x)\|^2 = \sum_{i=1}^n |f_i(x)|^2$.

Several different approaches have been proposed to solve the above $L1$-harmonic minimization problem. In [6, 7], (3.1) is tackled by iteratively solving its nonlinear Euler-Lagrange equation and renormalization. A penalty method is considered in [41] to penalize violation of the constraint with the following unconstrained objective functional $\mathcal{E}_\epsilon(\vec{F}) = \int_\Omega |\nabla \vec{F}| \mathrm{d}x + \frac{1}{\epsilon}(\|\vec{F}(x)\|^2 - 1)^2$. A third approach in [11] is based on solving the unconstrained problem $\int_\Omega \left| \nabla \left(\frac{\vec{F}}{|\vec{F}|} \right) \right| \mathrm{d}x$ using a gradient descent method. More recently, a new approach is introduced in [13] based on constraint preserving curvilinear search on the unit sphere with choosing Barzilai-Borwein step size for updating. The authors in [13] also claim the most efficiency of the curvilinear search method compared with the previous approaches. Thus, our comparison is mainly conducted with the curvilinear search method.

According to the discussion of the SOC method in section 2, we propose to solve the problem (3.1) using the Algorithm 1 by introducing an auxiliary variable $\vec{P}$ for $\vec{F}$, which solves the following subproblem:

$$\vec{F}^k = \arg\min_{\vec{F}} \int_\Omega |\nabla \vec{F}| + \frac{r}{2} \int_\Omega \|\vec{F} - \vec{P}^{k-1} + \vec{B}^{k-1}\|^2 \tag{3.2}$$

This unconstrained problem is a well-known Rudin-Osher-Fatemi (ROF) model [42], which can be solved by several well-known algorithms. Here, we use split Bregman method [38] to solve the problem (3.2) by

introducing another auxiliary variable $\vec{\vec{Q}}$ for $\nabla\vec{F}$. In other words, we iteratively solve the following problems:

$$(\vec{F}^k, \vec{\vec{Q}}^k) = \arg\min_{\vec{F},\vec{\vec{Q}}} \int_\Omega |\vec{\vec{Q}}| + \frac{r}{2}\int_\Omega \|\vec{F} - \vec{P}^{k-1} + \vec{B}^{k-1}\|^2 + \frac{\eta}{2}\int_\Omega \|\nabla\vec{F} - \vec{\vec{Q}} + \vec{\vec{D}}\|^2 \tag{3.3}$$

$$\vec{\vec{D}} \leftarrow \vec{\vec{D}} + \nabla\vec{F}^k - \vec{\vec{Q}}^k \tag{3.4}$$

Combining with solving (3.2) using the split Bregman method [38] and the Algorithm 1, we propose the following algorithm to solve the $L1$-harmonic minimization problem (3.1).

SOC Algorithm 4. *Initialize* $\vec{B}^0 = 0, \vec{\vec{D}}^0 = 0, \vec{F}^0 = \vec{P}^0$ *and* $\vec{\vec{Q}}^0 = \nabla\vec{F}^0$

while *"not converge"* **do**

1. *Solve* $(-\eta\Delta + r)\vec{F}^k = r(\vec{P}^{k-1} - \vec{B}^{k-1}) - \eta\nabla\cdot(\vec{\vec{Q}}^{k-1} - \vec{\vec{D}}^{k-1})$,
2. $\vec{\vec{Q}}^k = \max\{0, 1 - \frac{1}{\eta\left|\nabla\vec{F}^k + \vec{\vec{D}}^{k-1}\right|}\}\left(\nabla\vec{F}^k + \vec{\vec{D}}^{k-1}\right)$
3. $\vec{P}^k(x) = \mathcal{S}_{proj}(\vec{F}^k(x) + B^k(x))$,
4. $\vec{\vec{D}}^k = \vec{\vec{D}}^{k-1} + \nabla\vec{F}^k - \vec{\vec{Q}}^k$,
5. $\vec{B}^k = \vec{B}^{k-1} + \vec{F}^k - \vec{P}^k$.

Note that the only equation needs to be solved in this algorithm is the first step for updating $\vec{F}^k$. This linear equation can be efficiently solved by a Fast Fourier Transformation (FFT) or iteratively solved by conjugate gradient method or Gauss-Seidel method. Here, we choose several steps of Gauss-Seidel to obtain an approximate solution similar to what was used in the split Bregman method [38]. According to the error forgetting property of Bregman iteration for the convex l_1 related problems [33], we expect that similar behavior of fast convergence can be observed for the proposed algorithm.

Different from the theoretically guarantee of convergence in the convex case, it is a challenge to have the convergence analysis of the proposed algorithm for the orthogonality constrained problem. To demonstrate the robustness of the proposed SOC algorithm, we first would like to consider the $L1$-harmonic energy minimization problem for direction field correction using the Algorithm 4. More specifically, we typically consider a unit norm direction field $\vec{F}(x) = (f_1(x), f_2(x))$ defined on the domain $D = [-1,1]^2$ with Dirichlet boundary condition $\frac{x}{\|x\|} = \frac{1}{\sqrt{x_1^2 + x_2^2}}(x_1, x_2)$ on the boundary ∂D. We choose the same initial input $\vec{F}^0 : D = [-1,1]^2 \to S^2$ using in [13] defined as follows:

$$\vec{F}^0(x) = \begin{cases} \left(\frac{x}{\|x\|}\sin\phi(\|x\|), \cos\phi(\|x\|)\right), & \text{if} \quad x \in (-1,1)^2 \\ \frac{x}{\|x\|}, & \text{if} \quad x \in \partial D \end{cases} \tag{3.5}$$

where $\phi(\|x\|) = \frac{3\pi}{2}\min\{\|x\|^2, 1\}$. By choosing grid width of the domain $[-1,1]^2$ as $h = \sqrt{2}/2^n$, we set the maximal iteration numbers as 10000 and terminate the program if $\frac{\|\vec{F}^k - \vec{F}^{k-1}\|}{\|\vec{F}^k\|} \le 10^{-6}$ and $\frac{|\mathcal{E}(\vec{F}^k) - \mathcal{E}(\vec{F}^{k-1})|}{\mathcal{E}(\vec{F}^k)} \le 10^{-7}$. In this case, the solution of the optimization problem (3.1) is given by $\vec{F}_T(x) = \frac{x}{\|x\|}$. We further define the relative error of the computation results by $Error = \|\vec{F}^k - \vec{F}_T\|/\|\vec{F}_T\|$ to check the accuracy of the proposed algorithm. In addition, we also numerically test the dependence of the algorithm performance to the parameter r.

The convergence of Bregman iteration method has been proved for convex problems [31, 32, 33]. For the orthogonality constrained problems, the non-convexity may introduce further restriction to the choice of the

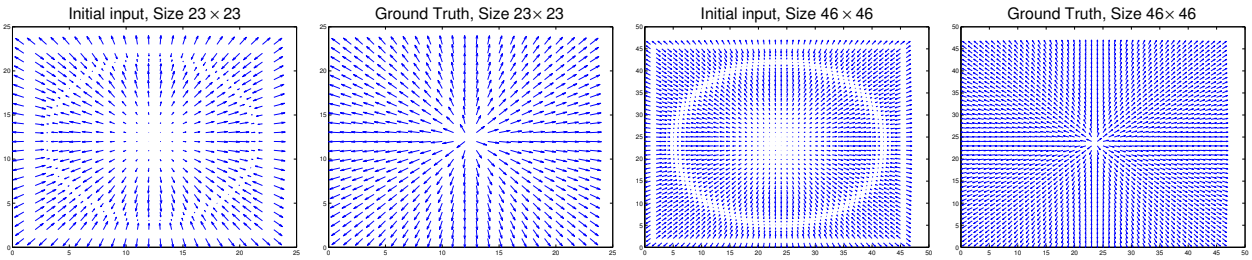


FIG. 3.1. *The initial input and ground truth of the problem with grid size* 23×23 *and* 46×46.

parameter r in the proposed algorithm. Intuitively, the larger r will force the stronger equality constraint between $\vec{F}$ and the auxiliary variable $\vec{P}$. While, a too much small r might not be able to provide strong enough force to drive $\vec{F}$ to satisfy the orthogonality condition. To demonstrate this point, we apply the Algorithm 4 to the above directional field correction problem with grid size 23×23 and 46×46 (Figure 3.1). Fixing $\mu = 50$, we choose a small $r = 1$ and relative big $r = 300$ in our fist test. Figure 3.2 reports the difference $\log \|\vec{F} - \vec{P}\|^2$, the difference $\log \|(\|\vec{F}(x)\|^2 - 1)\|^2$, the energy evolution via the number of iterations and the direction filed correction results. From the first and third columns of Figure 3.2, we can clearly see that the difference $\|\vec{F} - \vec{P}\|$ does not converge, thus the orthogonality constraint $\|\vec{F}(x)\|^2 = 1$ could not be satisfied and the computation results are not close to the ground truth. As comparisons, the second the fourth columns of Figure 3.2 illustrate the computation results using $r = 300$. It is clear to see that $\|\vec{F} - \vec{P}\|$ converge and the orthogonality constraint also satisfied. To further test the dependence of the algorithm performance to the parameter r, we also test the algorithm by choosing r from 1 to 1500. In Figure 3.3, we report the relative errors $\|\vec{F}^k - \vec{F}_T\|/\|\vec{F}_T\|$ and number of iterations using different values of r. According to the Figure. 3.3, we can also clearly see that the algorithm always converge as long as r is great than a certain value. In addition, the relative errors $\|\vec{F}^k - \vec{F}_T\|/\|\vec{F}_T\|$ does not depend on the values of r once the algorithm converge. While, the iteration numbers of the algorithm does depend on the parameter r. The bigger r is chosen, the more iterations the algorithm needs to run. Unfortunately, we currently could not theoretically indicate how to choose a suitable threshold of the parameter r to guarantee the algorithm convergence. This is certainly an interesting point to explore in our future work.

To further illustrate the efficiency of the proposed algorithm, we compare our algorithm with the curvilinear search algorithm proposed by Wen et al. [13, 28] recently, whose efficiency has been demonstrated to be much better than the methods introduced in [41, 11]. We test two direction fields with size 23×23 and 46×46 and the initial input defined in (3.5) for both algorithms and choose $r = 300, \eta = 50$ in the algorithm 4. Figure 3.4 plots numerical results and energy evolution curves obtained from both algorithms. The table in Figure 3.4 reports the number of iterations, computation time and the accuracy $\|\vec{F}^k - \vec{F}_T\|/\|\vec{F}_T\|$ for both algorithms. According to the comparison, it is clear to see that both algorithms can provide satisfactory results. The proposed SOC algorithm is slightly more expensive than the curvilinear search algorithm in terms of the computation time for each iteration, while the total iteration numbers of the SOC algorithm are much more fewer than the iteration numbers of the curvilinear search method. Thus, the total computation time of our method is much less than the total time of curvilinear search method. Moreover, our proposed algorithm 4 apparently is more accurate than the curvilinear search method for the $L1$-harmonic energy minimization problem.

Next, we apply the Algorithm 4 to solve the $L1$-harmonic energy minimization problem (3.1) in the case of $n = 3$ for an application to color RGB image chromaticity denoising [8, 9, 10, 11, 13]. In this

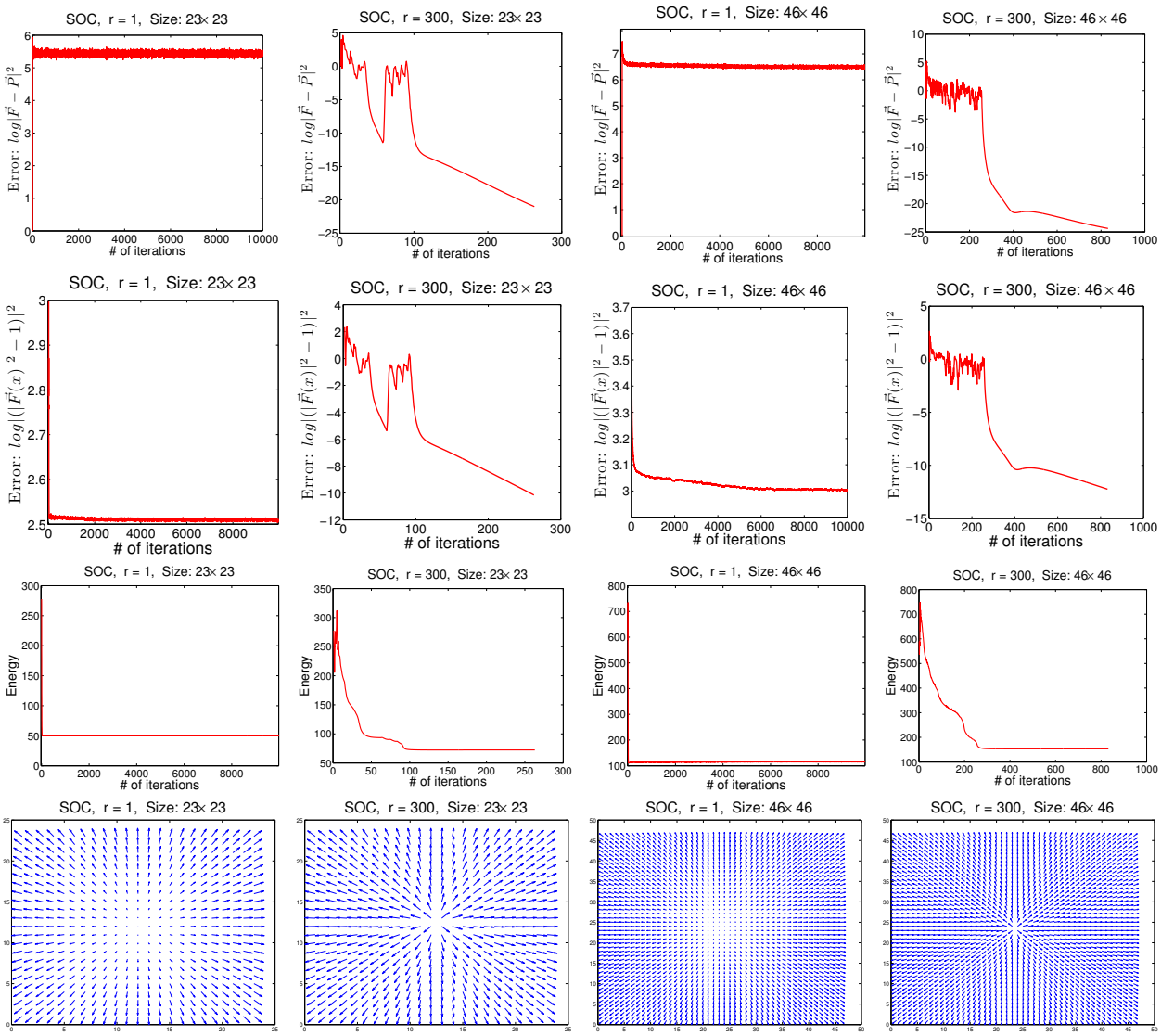


FIG. 3.2. *The algorithm performances by choosing different r. From top to bottom, each row plots the error $\log\|\vec{F}-\vec{P}\|^2$, the error $\log\|(|\vec{F}(x)|^2-1)\|^2$, the energy evolution curve and computing results, respectively. The first two columns report results by choosing grid size 23×23 and $r=1,500$ respectively. The last two columns report results by choosing grid size 46×46 and $r=1,500$ respectively.*

application, a color image is represented as $\vec{I}=(I_R,I_G,I_B)$, from which the brightness of the image is given by $|\vec{I}|=\sqrt{I_R^2+I_G^2+I_B^2}$ and the chromaticity is given by $\vec{F}=\dfrac{\vec{I}}{|\vec{I}|}=\dfrac{1}{\sqrt{I_R^2+I_G^2+I_B^2}}(I_R,I_G,I_B)\in S^2$. To test the L1-harmonic minimization for chromaticity denoising, we only assume that the chromaticity is contaminated but the brightness is kept from the original image. In our test, Gaussian noise is added to the chromaticity $\vec{F}$ so that the image with noisy chromaticity is given by $\vec{F}^0=\dfrac{\vec{F}+\beta\xi}{|\vec{F}+\beta\xi|}$, where ξ satisfies the standard normal distribution $N(0,1)$. With the solution $\vec{F}^*$ of the $L1$ harmonic flow guided by $L1$-harmonic energy, the denoised image can be assembled as $\vec{I}^*=|\vec{I}|\vec{F}^*$.

We test two color images, a flower image and a clown image, with two different levels of Gaussian noise, $\beta=0.4,0.8$, using the proposed Algorithm 4 and the curvilinear search method introduced in [13].

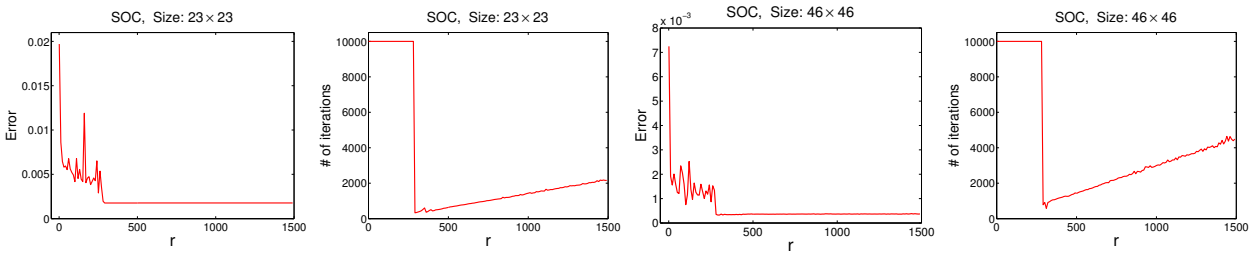
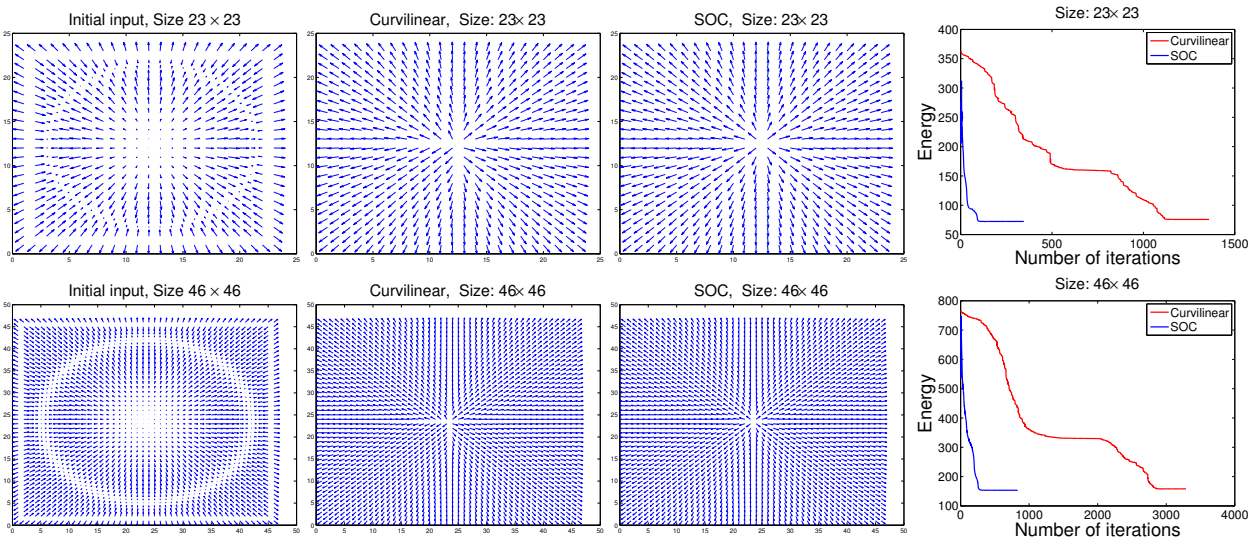


FIG. 3.3. *The relative error* $\|\vec{F}^k - \vec{F}_T\|/\|\vec{F}_T\|$ *and number of iterations via the parameter* r*. The first two plots report results using grid size* 23×23*. The last two plots report results using grid size* 46×46*.*



grid size	SOC Algorithm 4				the curvilinear algorithm			
	# of iterations	time(s)		*Error*	# of iterations	time(s)		*Error*
		per-iter	total			per-iter	total	
23×23	346	0.000675	0.23	0.0018	1359	0.000543	0.74	0.0038
46×46	832	0.001308	1.09	0.0004	3288	0.001001	3.29	0.0015

FIG. 3.4. *Comparison between the proposed algorithm 4 and the curvilinear search algorithm in [13, 28]. The first column shows the initial inputs* $\vec{F}^0$ *for both algorithms. The second and the third columns are results obtained from the curvilinear algorithm and our proposed SOC algorithm, respectively. The last column illustrates the energy evolution via iteration numbers. The table list the comparisons of numbers of iterations, computation time and accuracy of both algorithms.*

Figure. 3.5 plots resulting denoised images and corresponding energy evolution curves. The first and third rows of Figure. 3.5 plot results of noise level $\beta = 0.4$ and the second and fourth rows report results of noise level $\beta = 0.8$. The table in Figure 3.5 reports the caparisons of computation time and iteration numbers for both algorithms. Although the proposed SOC algorithm takes slightly more time than the curvilinear search method, the total iteration numbers of the SOC algorithm are much more fewer than the iteration numbers of the curvilinear search method. Overall, the SOC algorithm is more efficient than curvilinear search method in this problem.

3.2. Global conformal parameterization for genus-0 surfaces. Surface parameterization allows operations on the surface to be carried out on simple parameter domains. A special type of parameterization

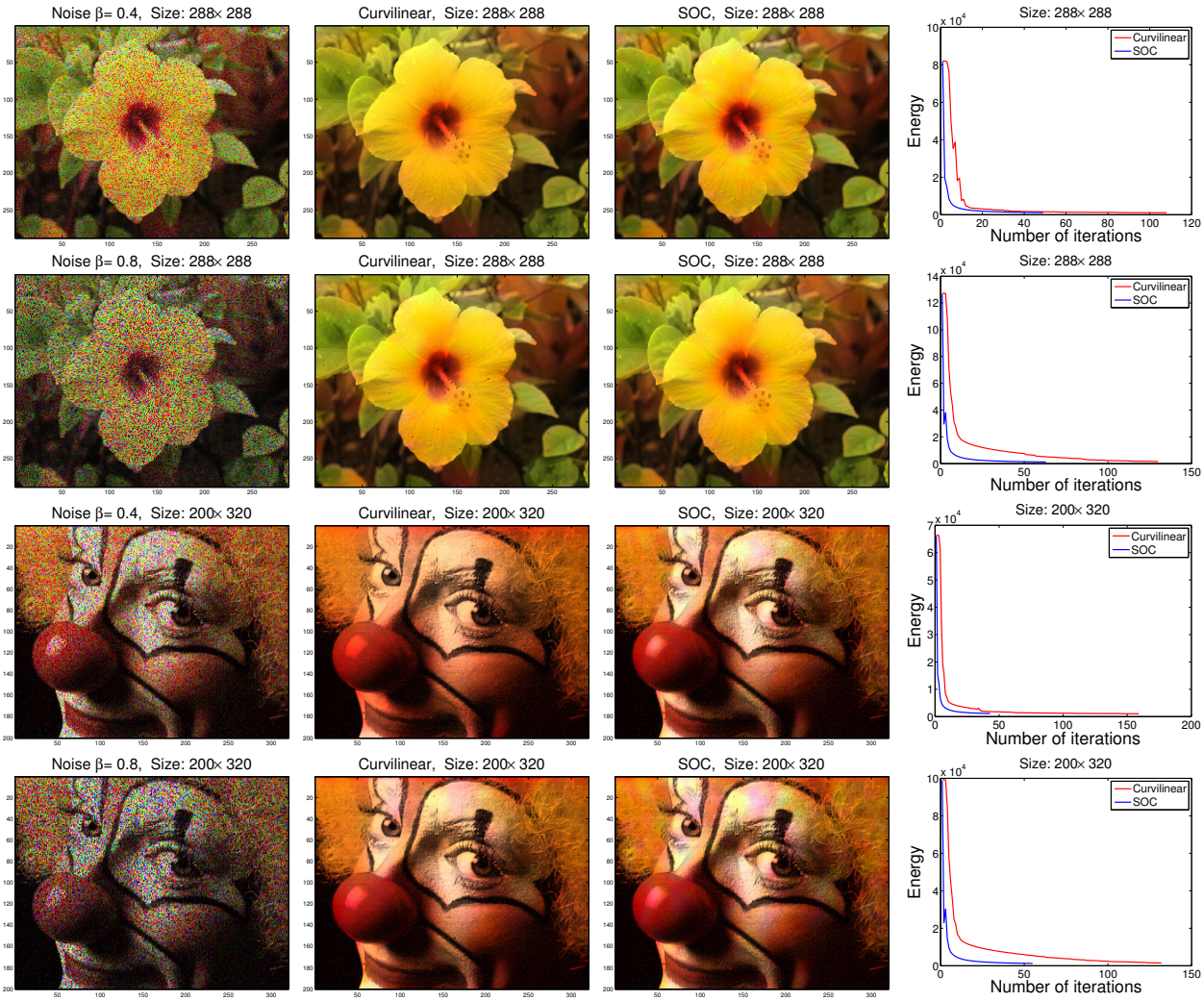


image	SOC Algorithm 4			the curvilinear algorithm[13]		
	# of iterations	time(s)		# of iterations	time(s)	
		per-iter	total		per-iter	total
flower ($\beta = 0.4$)	49	0.0758	3.72	108	0.0647	6.99
flower ($\beta = 0.8$)	63	0.0759	4.97	130	0.0622	8.10
clown ($\beta = 0.4$)	43	0.0479	2.06	159	0.0465	7.39
clown ($\beta = 0.8$)	55	0.0479	2.64	132	0.0449	5.92

FIG. 3.5. *Comparison between the proposed algorithm 4 and the curvilinear algorithm in [13] with fixed 30 iterations. The first column shows the input images contaminated by two different levels of Gaussian noise on their chromaticity. The second and the third columns are results obtained from the curvilinear algorithm and our proposed SOC algorithm 4, respectively. The last column illustrates the energy evolution via iteration numbers. The table list the comparisons of numbers of iterations, computation time of both algorithms.*

with the angle preserving property, so called *conformal parameterization*, is widely used in many different areas such as computer vision, computer graphics, as well as medical image analysis [43, 44, 45, 46, 1, 2, 47, 48, 49, 50]. In many situations, a *global* conformal parameterization that maps a surface onto a global parameter domain is desirable. Particularly, a global conformal parameterization of a genus-0 surface $(\mathcal{M}, g)$ is equivalent to a harmonic map from $\mathcal{M}$ to the unit sphere (S^2, g_0), which can be obtained by optimizing

the following harmonic/Dirichlet energy minimization problem [51, 52]:

$$\min_{\vec{F}=(f_1,f_2,f_3):\mathcal{M}\to\mathbb{R}^3} \mathcal{E}(\vec{F}) = \frac{1}{2}\int_{\mathcal{M}} \sum_{i=1}^{3} |\nabla_{\mathcal{M}} f_i(x)|^2 \mathrm{d}\mathcal{M}, \quad s.t. \quad f_1^2(x) + f_2^2(x) + f_3^2(x) = 1 \tag{3.6}$$

where $\nabla_{\mathcal{M}} f_\alpha = \sum_{i,j=1}^{2} g^{ij} \frac{\partial f_\alpha}{\partial x^i} \partial_{x^j}$ is gradient of $f_\alpha, \alpha = 1, 2, 3$ on $(\mathcal{M}, g)$ and (g^{ij}) is the inverse of the metric matrix g. If we write $\nabla_{\mathcal{M}} \vec{F} = (\nabla_{\mathcal{M}} f_1, \nabla_{\mathcal{M}} f_2, \nabla_{\mathcal{M}} f_3)$ and $\|\nabla_{\mathcal{M}} \vec{F}\|^2 = \sum_{i=1}^{3} |\nabla_{\mathcal{M}} f_i(x)|^2$, then the above problem can be written as:

$$\min_{\vec{F}=(f_1,f_2,f_3)} \mathcal{E}(\vec{F}) = \frac{1}{2}\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} \vec{F}\|^2, \quad s.t. \quad \|\vec{F}(x)\|^2 = 1 \tag{3.7}$$

This constrained optimization problem is tackled in [1, 2] using the gradient descent method and iteratively projecting back to the sphere. More recently, a more efficient algorithm for the above problem is discussed in [3] based on a curvilinear search method, with Barzilai-Borwein step size, introduced in [28].

Here, we can use the proposed Algorithm 1 to solve the problem (3.7) by introducing an auxiliary variable $\vec{P}$ for $\vec{F}$ to split the constraint, which is needed to solve the following subproblem:

$$\vec{F}^k = \arg\min_{\vec{F}} \frac{1}{2}\int_{\mathcal{M}} \|\nabla_{\mathcal{M}} \vec{F}\|^2 + \frac{r}{2}\int_{\mathcal{M}} \|\vec{F} - \vec{P}^{k-1} + \vec{B}^{k-1}\|^2 \tag{3.8}$$

whose minimizer satisfies the following Euler-Lagrangian equation:

$$(-\Delta_{\mathcal{M}} + r)\vec{F} = r(\vec{P}^{k-1} - \vec{B}^{k-1}) \tag{3.9}$$

where $\Delta_{\mathcal{M}}$ is the Laplace-Beltrami operator on $\mathcal{M}$ and $\Delta_{\mathcal{M}} \vec{F} = (\Delta_{\mathcal{M}} f_1, \Delta_{\mathcal{M}} f_2, \Delta_{\mathcal{M}} f_3)$. Therefore, the harmonic energy minimization problem (3.7) can be iteratively solved using our proposed SOC method described in Algorithm 1 as follows:

SOC Algorithm 5. *Initialize* $\vec{B}^0 = 0, \vec{F}^0 = \vec{P}^0 =$ *Gauss map* $\mathcal{G}$

while *"not converge"* **do**

1. *Solve* $(-\Delta_{\mathcal{M}} + r)\vec{F}^k = r(\vec{P}^{k-1} - \vec{B}^{k-1})$,
2. $\vec{P}^k(x) = \mathcal{S}_{proj}(\vec{F}^k(x) + \vec{B}^{k-1}(x)) = \dfrac{\vec{F}^k(x) + \vec{B}^{k-1}(x)}{|\vec{F}^k(x) + \vec{B}^{k-1}(x)|}$,
3. $\vec{B}^k = \vec{B}^{k-1} + (\vec{F}^k - \vec{P}^k)$.

Where the Gauss map $\mathcal{G} : \mathcal{M} \to S^2, \mathcal{G}(p) = \vec{n}_p$, in which $\vec{n}_p$ is the unit outward normal vector at $p \in \mathcal{M}$. Comparing with the curvilinear method in [28, 3], the algorithm is very easy to implement without requiring further sophisticated optimization techniques. Meanwhile, it has comparable efficiency with the curvilinear method, which will be illustrated in the numerical experiments.

In practice, we approximate $\mathcal{M}$ by a triangulated surface $\mathcal{M} = \{V = \{p_i\}_{i=1}^N, T = \{T_l\}_{l=1}^L\}$, where $p_i \in \mathbb{R}^3$ is the i-th vertex and T_l is the l-th triangle. For a function $f = (f(p_1), \cdots, f(p_N))^T$ defined on the triangle mesh, we approximate the Laplace-Beltrami operator on surface $\mathcal{M}$ by [53, 54, 55]:

$$\triangle_{\mathcal{M}} f(p_i) \approx \sum_{j\in N_i} \omega_{ij}(p_i)\left(f(p_j) - f(p_i)\right), \tag{3.10}$$

where $\omega_{ij}(p_i) = \dfrac{cot\alpha_{ij}(p_i) + cot\beta_{ij}(p_i)}{2}$, α_{ij} and β_{ij} are the two angles opposite to the edge $\overline{p_i p_j}$, N_i is the first ring neighborhood of the vertex p_i.

Since the discretization of the operator $(-\Delta_{\mathcal{M}} + r)$ is a sparse and symmetric positive definite matrix, there are variety of numerical linear algebra packages to solve the equation (3.9), such as Jacobi, Gauss-Seidel, or conjugate gradient method etc. In all our numerical experiments, 5 steps of Gauss-Seidel are chosen to approximate the solution of (3.9).

Given a general genus-0 surface $(\mathcal{M}, g)$, it is hard to have analytical forms of the conformal maps from $(\mathcal{M}, g)$ to (S^2, g_0). However, the angle preserving property and the relation of conformal factor $\vec{F}^*(g_0) = e^{2u} g$ [1] can help us check the accuracy of the resulting conformal map $\vec{F} : \mathcal{M} \to S^2, p \mapsto (f_1(p), f_2(p), f_3(p))$ by checking its conformality. Numerically, we can compute the angle differences between triangles on the input surface $\mathcal{M}$ and the corresponding triangles on the obtained map. In addition, the "conformal factor" with respect to the map $\vec{F}$ can be approximated by:

$$e^{2u(p_i)} = \frac{\sum_{p_i \in T_l} \text{Area}(\vec{F}(T_l))}{\sum_{p_i \in T_l} \text{Area}(T_l)} \tag{3.11}$$

To further illustrate the efficiency of the proposed Algorithm 5, we conduct comparisons of our algorithm with the gradient descent approach in [2] and curvilinear search method in [28, 3]. For all three algorithms, surface Gauss maps are chosen to be the initial maps and the stop criteria is chosen as $\frac{|\mathcal{E}(\vec{F}^k) - \mathcal{E}(\vec{F}^{k-1})|}{\mathcal{E}(\vec{F}^k)} \leq \epsilon = 10^{-10}$. Figure 3.2 plots the resulting conformal factors, angle differences and curves of energy evolution for three algorithms. The numbers of iterations and corresponding computation time are reported in Table 3.2. According to the corresponding patterns of the conformal factors and angle differences illustrated in Figure 3.2, all three methods provide satisfactory results. It is clear to see that both of the proposed method and the curvilinear method are much more efficient than the gradient descent method. Our proposed algorithm based on the SOC method has the comparable efficiency with the curvilinear search method. Again, we would like to point out that our proposed method is more convenient to implement than the curvilinear search method which needs more attention about Cayley transformation and Barzilai-Borwein step size.

Surface	# of verticies	SOC Algorithm 5		the curvilinear algorithm		the gradient descent method	
		# of iterations	time(s)	# of iterations	time(s)	# of iterations	time(s)
Hippo1	2000	362	0.34	210	0.19	3994	2.08
Putamen1	10002	1036	4.57	1157	4.86	30000	89.74

TABLE 3.1

Comparison of the proposed Algorithm 5 with the curvilinear algorithm in [28, 3] and the gradient descent method in [1].

4. Conclusion and Future Work. In this paper, we tackle the orthogonality constrained problems using the proposed SOC method. With the introduced auxiliary variables, the SOC method iteratively solves the constrained problem by computing an unconstrained problem and a quadratic constrained problem with an analytic solution. Different from penalty methods and the standard augmented Lagrangian methods, the SOC method directly solves the orthogonality constraint in each iteration. Meanwhile, it also successfully avoids handling the complicated manifold structure as the feasible approaches need to do. To demonstrate the robustness of the SOC method, three applications, conformal mapping construction, directional field

[1] $\vec{F}^*$ is the standard *pull-back* map in differential geometry.

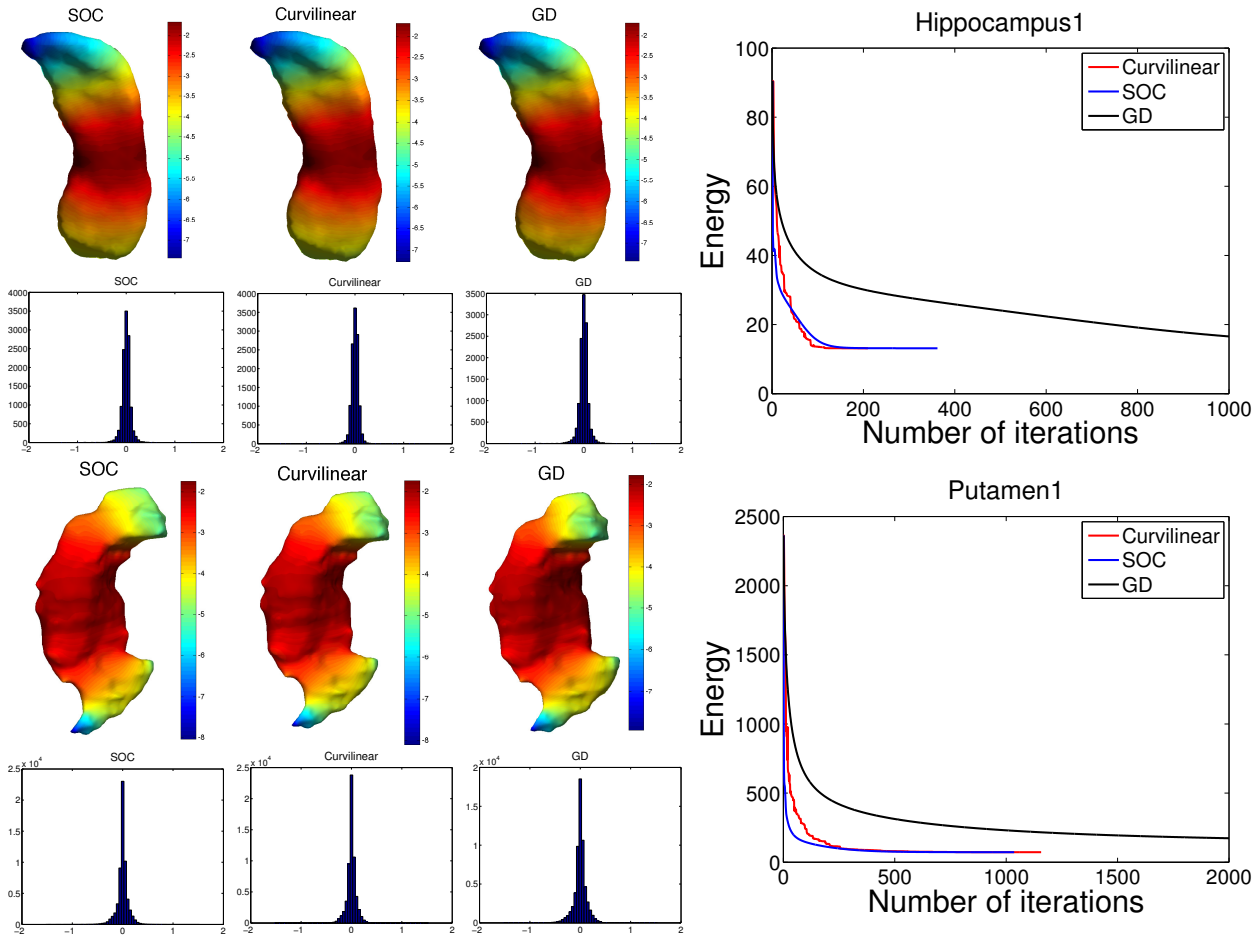


FIG. 3.6. *Comparison the proposed Algorithm 5 with the curvilinear algorithm in [28, 3] and the gradient descent method in [2]. Surfaces are color coded by the conformal factors obtained from the resulting conformal mappings. Histograms show angle difference between triangles on the input surface $\mathcal{M}$ and the corresponding triangles on the obtained map. The last column illustrates the energy evolution via iteration numbers.*

correction and color image restoration, are discussed. In addition, we also conduct comparisons of the SOC method with existing fast algorithms for orthogonality constrained problems to illustrate its efficiency. According to our numerical comparison, the efficiency of the SOC method is comparable with curvilinear search method in conformal mapping construction problem. Moreover, the SOC method converges much faster than the curvilinear search method for l_1 related problems due to the error forgetting of Bregman iteration.

Our applications and numerical experiments focus on spherical constrained problems so far, although we also propose algorithms to solve matrix orthogonality constrained problems and spherical and linear equality constrained problems. The applications and numerical demonstrations of the latter two types of problems will be explored in our future work. More importantly, we have not been able to show the convergence of the SOC method in this paper, which we believe it is true according to our experiments. In our future work, we will certainly take the proof of convergence of the SOC method as a high priority.

Acknowledgement. Rongjie Lai's work is supported by Zumberge Individual Award from USC's James H. Zumberge Faculty Research and Innovation Fund. Stanley Osher's work is supported by NSF grant DMS-1118971, DMS-0914561, ONR grant N00014-08-1-1119, an ARO MURI subcontract from University

of South Carolina and an ARO MURI subcontract from Rice University. The authors would like to express their gratitude to Prof. Zaiwen Wen and Prof. Wotao Yin for sharing their curvilinear search codes for numerical comparisons.

REFERENCES

[1] X. Gu and S. Yau. Computing conformal structures of surfaces. *Communications in Information and Systems*, 2(2):121–146, 2002.

[2] X. Gu and S. T. Yau. Global conformal surface parameterization. *Symposium on Geometry Processing*, pages 127–137, 2003.

[3] R. Lai, Z. Wen, W. Yin, X. Gu, and L. M. Lui. Folding-free global conformal mapping for genus-0 surfaces by harmonic energy minimization. *in revision, Journal Scientific Computing*, 2013.

[4] P. Perona. Orientation diffusion. *IEEE Trans. Image Process.*, 79:12–49, 1998.

[5] S.-Y. Lin and M. Luskin. Relaxation methods for liquid crystal problems,. *SIAM J. Numer. Anal.*, 26:1310–1324, 1989.

[6] R. Cohen, R. Hardt, D. Kinderlehrer, S. Y. Lin, and M. Luskin. Minimum energy configurations for liquid crystals: Computational results. *Theory and Applications of Liquid Crystals (Minneapolis, MN, 1985), IMA Vol. Math. Appl. 5*, pages 99–121, 1987.

[7] R. Cohen, S. Y. Lin, and M. Luskin. Relaxation and gradient methods for molecular orientation in liquid crystals. *Comput. Phys. Comm.*, 53:455—465, 1989.

[8] T. F. Chan and J. Shen. Variational restoration of nonflat image features: Models and algorithms. *SIAM J. Appl. Math.*, 61:1338–1361, 2000.

[9] B. Tang, G. Sapiro, and V. Caselles. Diffusion of general data on non-flat manifolds via harmonic maps theory: The direction diffusion case. *Int. J. Comput. Vision*, 36:149–161, 2000.

[10] B. Tang, G. Sapiro, and V. Caselles. Color image enhancement via chromaticity diffusion. *IEEE Trans. Image Process.*, 10:701–707, 2001.

[11] Luminita A. Vese and Stanley J. Osher. Numerical methods for p-harmonic flows and applications to image processing. *SIAM J. Numer. Anal.*, 40(6):2085–2104, 2002.

[12] Martin Kružík and Andreas Prohl. Recent developments in the modeling, analysis, and numerics of ferromagnetism. *SIAM Rev.*, 48(3):439–483, 2006.

[13] Donald Goldfarb, Zaiwen Wen, and Wotao Yin. A curvilinear search method for the p-harmonic flow on spheres. *SIAM Journal on Imaging Sciences*, 2:84–109, 2009.

[14] P.T. Boufounos and R.G. Baraniuk. 1-bit compressive sensing. In *Conference on Information Sciences and Systems (CISS)*, pages 16–21. IEEE, 2008.

[15] J.N. Laska, Z. Wen, W. Yin, and R.G. Baraniuk. Trust, but verify: Fast and accurate signal recovery from 1-bit compressive measurements. *IEEE Trans. Signal Process.*, 59(11):5289, 2011.

[16] M. Yan, Y. Yang, and S. Osher. Robust 1-bit compressive sensing using adaptive outlier pursuit. *IEEE Trans. Signal Process.*, 60(7):3868–3875, 2012.

[17] A. Edelman, T. As, A. Arias, T. Smith, et al. The geometry of algorithms with orthogonality constraints. *SIAM J. Matrix Anal. Appl*, 20(2):303–353, 1998.

[18] A Caboussat, R Glowinski, and V Pons. An augmented lagrangian approach to the numerical solution of a non-smooth eigenvalue problem. *Journal of Numerical Mathematics*, 17(1):3–26, 2009.

[19] R.E. Burkard, S.E. Karisch, and F. Rendl. Qaplib–a quadratic assignment problem library. *Journal of Global Optimization*, 10(4):391–403, 1997.

[20] E.M. Loiola, N.M.M. de Abreu, P.O. Boaventura-Netto, P. Hahn, and T. Querido. A survey for the quadratic assignment problem. *European Journal of Operational Research*, 176(2):657–690, 2007.

[21] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, 2006.

[22] F. Bethuel, H. Brezis, and F. Hélein. Asymptotics for the minimization of a ginzburg-landau functional. *Calculus of Variations and Partial Differential Equations*, 1(2):123–148, 1993.

[23] R. Glowinski and P. Le Tallee. *Augmented Lagrangian and Operator-Splitting Methods in Nonlinear Mechanics*. SIAM, 1989.

[24] M. Fortin and R. Glowinski. *Augmented Lagrangian methods: applications to the numerical solution of boundary-value problems*, volume 15. North Holland, 2000.

[25] J.H. Manton. Optimization algorithms exploiting unitary constraints. *IEEE Trans. Signal Process.*, 50(3):635–650, 2002.

[26] I. Yamada and T. Ezaki. An orthogonal matrix optimization by dual cayley parametrization technique. In *4th International Symposium on Independent Component Analysis and Blind Signal Separation (ICA2003)*, 2003.

[27] P.-A. Absil, R. Mahony, and R. Sepulchre. *Optimization algorithms on matrix manifolds*. Princeton University Press, Princeton, NJ, 2008.

[28] Z. Wen and W. Yin. A feasible method for optimization with orthogonality constraints. *UCLA CAM 10-77*, 2010.

[29] V. Estellers, D. Zosso, R. Lai, J.P. Thiran, S. Osher, and X. Bresson. An efficient algorithm for level set method preserving distance function. *In press, IEEE Trans. Image Process.*, 2012.

[30] R. Glowinskiand M. Holmstrom. Constrained motion problems with applications by nonlinear programming methods. *Surveys on Math. for Industry*, 5:75–108, 1995.

[31] Stanley Osher, Martin Burger, Donald Goldfarb, Jinjun Xu, and Wotao Yin. An iterative regularization method for total variation-based image restoration. *Multiscale Model. Simul.*, 4(2):460–489, 2005.

[32] Wotao Yin, Stanley Osher, Donald Goldfarb, and Jerome Darbon. Bregman iterative algorithms for ℓ_1-minimization with applications to compressed sensing. *SIAM Journal on Imaging Sciences*, 1(1):143–168, 2008.

[33] W. Yin and S. Osher. Error forgetting of bregman iteration. *UCLA CAM Report (12-08)*, 2012.

[34] L. Bregman. The relaxation method of nding the common points of convex sets and its application to the solution of problems in convex optimization. *USSR Computational Mathematics and Mathematical Physics*, 7:200–217, 1967.

[35] E. Esser. Applications of lagrangian-based alternating direction methods and connections to split bregman. *UCLA CAM Report (09-31)*, 2009.

[36] X. Tai and C. Wu. Augmented lagrangian method, dual methods and split bregman iteration for rof model. *SSVM 09 Proceedings of the Second International Conference on Scale Space and Variational Methods in Computer Vision*, 2009.

[37] R. Glowinski and A. Marrocco. Sur lapproximation par elements finis dordre un, et la resolution par penalisation-dualite dune classe de problemes de dirichlet nonlineaires. *Rev. Francaise dAut. Inf. Rech. Oper.*, R-2:41–76, 1975.

[38] T. Goldstein and S. Osher. The split bregman method for l1 regularized problems. *UCLA CAM Report 08-29*, 2008.

[39] G. Rosman, Y. Wang, X. Tai, R. Kimmel, and A. M. Bruckstein. Fast regularization of matrix-valued images. In *Computer Vision–ECCV 2012*, pages 173–186. Springer, 2012.

[40] WA Gibson. On the least-squares orthogonalization of an oblique transformation. *Psychometrika*, 27(2):193–195, 1962.

[41] F. Bethuel, H. Brezis, and F. Helein. Asymptotics for the minimization of a ginzburg-landau functional. *calculus of variations and partical differential euqations*, 1(2):123–148, 1993.

[42] L. Rudin, S. Osher, and E. Fatemi. Nonlinear total variation based noise removal algorithms. *Physica. D.*, 60:259–268, 1992.

[43] M. Eck, T. DeRose, T. Duchamp, H. Hoppe, M. Lounsbery, and W. Stuetzle. Multiresolution analysis of arbitrary meshes. *Proceeding of ACM SIGGRAPH*, 1995.

[44] T. Kanai, H. Suzuki, and F. Kimura. Three-dimensional geometric metamorphosis based on harmonic maps. *The Visual Computer*, 14(4):166–176, 1998.

[45] B. Levy, S. Petitjean, N. Ray, and J. Maillot. Least squares conformal maps for automatic texture atlas generation. *Proceeding of ACM SIGGRAPH*, 2002.

[46] P. Alliez, M. Meyer, and M. Desbrun. Interactive geometry remeshing. *Proceeding of ACM SIGGRAPH*, 2002.

[47] X. Gu, Y. Wang, T. F. Chan, P. Thompson, and S. T. Yau. Genus zero surface conformal mapping and its application to brain surface mapping. *IEEE Transaction on Medical Imaging*, 23:949–958, 2004.

[48] B. Springborn, P. Schröder, and U. Pinkall. Conformal equivalence of triangle meshes. *ACM Transactions on Graphics (TOG) - Proceedings of ACM SIGGRAPH 2008*, 27(3), 2008.

[49] L.M. Lui, Y. Wang, P. M. Thompson, and T. F. Chan. Landmark constrained genus zero surface conformal mapping and its application to brain mapping research. *Applied Numerical Mathematics*, 57:847–858, 2007.

[50] L.M. Lui, X. Gu, T.F. Chan, and S.-T. Yau. Variational method on riemann surfaces using conformal parameterization and its applications to image processing. *Methods and Applications of Analysis*, 15(4):513–538, 2008.

[51] R. Schoen and S.-T. Yau. *Lectures on Harmonic Maps*. International Press, 1994.

[52] J. Jost. Riemannian geometry and geometric analysis. *Springer, 3rd edition*, 2001.

[53] M. Meyer, M. Desbrun, P. Schröder, and A. H. Barr. Discrete differential-geometry operators for triangulated 2-manifolds. *Visualization and Mathematics III. (H.C. Hege and K. Polthier, Ed.) Springer Verlag*, pages 35–57, 2003.

[54] M. Desbrun, M. Meyer, P. Schröder, and A. H. Barr. Discrete differential geometry operators in nd. *In Proc. VisMath'02 Berlin Germany*, 2002.

[55] Guoliang Xu. Convergent discrete laplace-beltrami operator over triangular surfaces. *Proceedings of Geometric Modelling and Processing*, pages 195–204, 2004.