# Grouped Coordinate Descent Algorithms for Robust Edge-Preserving Image Restoration

Jeffrey A. Fessler
4240 EECS Bldg., University of Michigan, Ann Arbor, MI 48109-2122

**ABSTRACT**

We present a new class of algorithms for edge-preserving restoration of piecewise-smooth images measured in non-Gaussian noise under shift-variant blur. The algorithms are based on minimizing a regularized objective function, and are guaranteed to monotonically decrease the objective function. The algorithms are derived by using a combination of two previously unconnected concepts: A. De Pierro's convexity technique for optimization transfer, and P. Huber's iteration for M-estimation. Convergence to the unique global minimum is guaranteed for strictly convex objective functions. The convergence rate is very fast relative to conventional gradient-based iterations. The proposed algorithms are flexibly parallelizable, and easily accommodate nonnegativity constraints and arbitrary neighborhood structures. Implementation in Matlab is remarkably simple, requiring no cumbersome line searches or tolerance parameters.

**Keywords:** Image restoration, non-Gaussian noise, deconvolution, Bayesian methods.

## I. INTRODUCTION

In a wide variety of inverse problems one attempts to estimate an unknown object (e.g. an image) from a noisy measurement vector $\underline{y} \in \mathbb{R}^{m_1}$ under the following linear additive noise model

$$\underline{y} = \boldsymbol{A}\underline{x} + \text{noise}, \tag{1}$$

where $\underline{x} \in \mathbb{R}^p$ represents image pixel or voxel values, or a parameterization thereof (e.g. tensor product B-splines [1] or coefficients of other bases [2,3].) The matrix $\boldsymbol{A}$ represents the system response function, which is usually assumed known. In practice portions of $\boldsymbol{A}$ may be measured or modeled and therefore may have both deterministic and random uncertainties.

Since (1) suggests $\underline{y} \approx \boldsymbol{A}\underline{x}$, it is natural to try to estimate $\underline{x}$ by minimizing a function that encourages the estimate $\hat{\underline{x}}$ to fit to the data. When the measurement errors are independent, a natural cost function is the following:

$$J^{\text{data}}(\underline{x}) = \sum_{i=1}^{m_1} \psi_i^{\text{data}}([\boldsymbol{A}\underline{x} - \underline{y}]_i), \tag{2}$$

where

$$[\boldsymbol{A}\underline{x} - \underline{y}]_i = \sum_{j=1}^{p} a_{ij} x_j - y_i .$$

In the often-assumed case of zero-mean Gaussian noise with covariance $\boldsymbol{K} = \sigma^2 \boldsymbol{I}_{m_1}$, where $\boldsymbol{I}_n$ is the $n \times n$ identity matrix, $J^{\text{data}}(\cdot)$ is usually taken to be the negative log-likelihood of the measurements:

$$J^{\text{data}}(\underline{x}) = \frac{1}{2}(y - \boldsymbol{A}\underline{x})' \boldsymbol{K}^{-1} (y - \boldsymbol{A}\underline{x}) = \sum_{i=1}^{m_1} \psi([\boldsymbol{A}\underline{x} - \underline{y}]_i),$$

where $\psi(t) = \frac{1}{2} t^2/\sigma^2$, and where "$'$" denotes matrix/vector transpose. The latter objective function is not robust to outliers in the data, i.e., to deviations from the Gaussian noise assumption. In this paper we consider the more general case (2), which allows for nonquadratic functions, such as the Huber function used in M-estimation [4, p. 177]:

$$\psi(t) = \begin{cases} t^2/2, & |t| \le \delta \\ \delta|t| - \delta^2/2, & |t| > \delta. \end{cases} \tag{3}$$

Email: fessler@umich.edu, WWW: http://www.eecs.umich.edu/~fessler, Voice: 313-763-1434, FAX: 313-764-8041.

In addition, in (2) we allow the data-fit functions $\psi_i^{\text{data}}$ to depend on $i$, for both the sake of generality and specifically to allow for different weights for different measurements, which is important for heteroscedastic data such as in emission computed tomography and photon-limited optical imaging [5,6].

Solutions based solely on a data-fit measure such as (2) are generally unsatisfactory due to the ill-posed nature of most inverse problems. A popular remedy is to include in the objective function a penalty term that encourages the estimate to adhere to prior expectations such as piecewise smoothness. Most objective functions with penalty terms are special cases of the following general form:

$$\Phi(\underline{x}) = J^{\text{data}}(\underline{x}) + \sum_{i=1}^{m_2} \psi_i^{\text{penalty}}([\boldsymbol{C}\underline{x} - \underline{z}]_i), \tag{4}$$

where one can think of $\boldsymbol{C} \in \mathrm{I\!R}^{m_2 \times p}$ as inducing a collection of $m_2$ "soft constraints", namely $\boldsymbol{C}\underline{x} \approx \underline{z}$. We allow the dependence on $i$ in (4) both for the sake of generality and specifically to allow for different weights in different terms (e.g., in imaging problems one often uses a weight 1 for horizontal and vertical neighboring pixels and $1/\sqrt{2}$ for diagonal neighbors). This generality is also needed for the shift-variant penalty function described in [7].

A concrete example is the first-order quadratic roughness penalty with a first-order neighborhood, e.g. [5]. In this case, $\underline{z}$ is just the zero vector, and $\psi_i(t) = \beta t^2/2$ for some positive global regularization parameter $\beta$. The matrix $\boldsymbol{C}$ has the following representation. Let $\boldsymbol{D}_n$ denote the $(n-1) \times n$ 1st-order differencing matrix:

$$\boldsymbol{D}_n = \begin{bmatrix} 1 & -1 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 \\ & & \ddots & \ddots & \\ 0 & 0 & 0 & 1 & -1 \end{bmatrix}.$$

Then, for the first-order roughness penalty, $\boldsymbol{C}$ is the following $m_2 \times p$ matrix, where $m_2 = n_y(n_x - 1) + n_x(n_y - 1)$:

$$\boldsymbol{C} = \begin{bmatrix} \boldsymbol{D}_{n_x} \otimes \boldsymbol{I}_{n_y} \\ \boldsymbol{I}_{n_x} \otimes \boldsymbol{D}_{n_y} \end{bmatrix}$$

where $\otimes$ denotes Kronecker matrix product. Note that $\boldsymbol{C}'\boldsymbol{C}$ is a Toeplitz-block-Toeplitz matrix with kernel

$$\begin{bmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{bmatrix}.$$

The more general formulation (4) easily accommodates arbitrarily large neighborhoods. Also, to avoid oversmoothing edges, we allow for the $\psi_i$'s to be nonquadratic.

To simplify notation and to enable considerable generality, we unify the notation for the data-fit and penalty terms in (2) and (4). The remainder of this paper considers objective functions having the following very general form:

$$\Phi(\underline{x}) = \sum_{i=1}^{m} \psi_i([\boldsymbol{B}\underline{x} - \underline{c}]_i). \tag{5}$$

The specific case (4) corresponds to $m = m_1 + m_2$ with

$$\boldsymbol{B} = \begin{bmatrix} \boldsymbol{A} \\ \boldsymbol{C} \end{bmatrix} \in \mathrm{I\!R}^{m \times p}, \quad \underline{c} = \begin{bmatrix} \underline{y} \\ \underline{z} \end{bmatrix} \in \mathrm{I\!R}^{m}.$$

Our goal then is to estimate (i.e. reconstruct/restore) an image $\underline{x} \in \mathrm{I\!R}^p$ by finding a minimizer of $\Phi(\cdot)$:

$$\hat{\underline{x}} = \arg\min_{\underline{x}} \Phi(\underline{x}) \quad \text{or} \quad \hat{\underline{x}} = \arg\min_{\underline{x} \geq \underline{0}} \Phi(\underline{x}).$$

A very large number of problems are special cases of the above formulation. However, when some or all of the $\psi_i$'s are non-quadratic it can be challenging to find suitable algorithms for minimizing $\Phi$, even in the case considered here where $\Phi$ is convex. Generic optimization methods may be suboptimal for this problem with nonquadratic $\psi_i$'s since they do not exploit the structure of (5).

In this paper, we first describe the assumptions on $\boldsymbol{B}$ and the $\psi_i$'s that are used throughout the paper, and then review the successive-substitutions algorithm proposed by Huber [4]. Unfortunately Huber's algorithm does not accommodate a nonnegativity constraint and is impractical for imaging problems. Therefore, we propose a simple, practical, parallelizable algorithm for minimizing (5) that is a generalization of [8]. Convergence issues are discussed.

## II. ASSUMPTIONS

In this section we define the restrictions on the $\psi_i$ functions and the matrix $\boldsymbol{B}$ that are used to ensure that the algorithms are well defined and to establish global convergence. We conjecture that the algorithms would have local convergence under weaker conditions.

### *A. The $\psi_i$ functions*

We restrict attention to $\psi_i$'s that satisfy the following conditions (cf the conditions of Lemma 8.3 on p. 184 of [4]):

C1. $\psi$ is symmetric,
C2. $\psi$ is everywhere differentiable (and therefore continuous),
C3. $\dot{\psi}(t) = d/dt\, \psi(t)$ is non-decreasing,
C4. $\omega_\psi(t) = \dot{\psi}(t)/t$ is non-increasing for $t \geq 0$.
C5. $\omega_\psi(0) = \lim_{t\to 0} \dot{\psi}(t)/t$ is finite and nonzero, i.e. $0 < \omega_\psi(0) < \infty$,

For edge-preserving reconstruction and for robustness to non-Gaussian noise, one will often choose $\psi$'s for which $\omega_\psi$ is in fact *strictly decreasing* for $t \geq 0$. However, we want the proofs to also include the case $\psi(t) = \frac{1}{2}t^2$, for which $\omega_\psi(t) = 1$. Fortunately, the less-restrictive condition C4 is sufficient for convergence.

C1 is natural for symmetric noise distributions and for roughness penalty functions. C2 is needed for gradient-based iterations. C3 ensures convexity of $\Phi$. C5 ensures that the proposed iterations are well defined, and C4 is central to the convergence proofs.

A large class of $\psi$ functions satisfy the above conditions. One notable exception is the Generalized Gaussian class [9] $\psi(t) = |t|^p$ for $p < 2$, due to either C2 (for $p \leq 1$) or C5 (for $1 < p < 2$). Newton-type methods seem to be poorly suited to that class, and line-search methods appear necessary. Another exception is the "entropy" functions of the form $\psi(t) = t \log t$. Also excluded is $\psi(t) = |t|^p$ for $p > 2$, due to C4.

Note that our assumptions do not require twice-differentiability of $\psi$. Thus the proposed iterations can be applied to functions such as the Huber function (3) which cannot be used with standard Newton-Raphson iterations since the Huber function is not everywhere twice differentiable.

The following properties follow immediately from the above conditions.

P1. From C1 and C2, $\dot{\psi}(0) = 0$.
P2. From C3 and C5, $\dot{\psi}(t) \neq 0$ for $t \neq 0$. Thus $\psi$ cannot have any "flat segments."
P3. From C1 and C3, $\psi$ is convex. (See p. 114 of [10].)
P4. From C1, $\dot{\psi}(-t) = -\dot{\psi}(t)$.
P5. From C3 and P4, if $|u| < |v|$, then $|\dot{\psi}(u)| \leq |\dot{\psi}(v)|$.
P6. From C1 and C5, $\omega_\psi(-t) = \omega_\psi(t)$.
P7. From C1 and C5, $\psi$ is twice differentiable at $t = 0$, and $\ddot{\psi}(0) = \omega_\psi(0)$, where $\ddot{\psi}$ is the second derivative of $\psi$.
P8. From C3, C5, C4, and P1, $0 < \omega_\psi(t) \leq \omega_\psi(0), \forall t$.
P9. If $\psi$ is twice differentiable at $t$, then by C4, $0 \geq \dot{\omega}_\psi(t) = (t\ddot{\psi}(t) - \dot{\psi}(t))/t^2$. Thus combining with C3, P8, and C5: $0 \leq \ddot{\psi}(t) \leq \omega_\psi(t) \leq \omega_\psi(0) < \infty$. In particular, if $\psi$ is everywhere twice differentiable, then the curvature of $\psi$ is bounded.
P10. Since $d/dt\, \psi(\sqrt{t}) = \dot{\psi}(\sqrt{t})/(2\sqrt{t}) = \omega_\psi(\sqrt{t})/2$ is nonincreasing for $t \geq 0$ by C4, the function $\psi(\sqrt{t})$ is concave for $t \geq 0$. (See p. 114 of [10].)

### *B. Convexity*

We assume that the matrix $\boldsymbol{B}$ (i.e. $\boldsymbol{A}$ and $\boldsymbol{C}$) has been designed to have full column rank. Equivalently,

$$\boldsymbol{B}' \operatorname{diag}\{w_i\} \, \boldsymbol{B} > \boldsymbol{0}, \tag{6}$$

i.e. is positive definite, for any positive values $\{w_i\}_{i=1}^m$, where $\operatorname{diag}\{w_i\}$ is the $m \times m$ diagonal matrix with $(i,i)$th entry $w_i$. A necessary condition therefore is that $m \geq p$. Assumption (6) is reasonable since eliminating zero and near-zero eigenvalues of $\boldsymbol{A}'\boldsymbol{A}$ is one of the main reasons for regularization in the first place. Indeed, typically the null space of $\boldsymbol{C}$ is simply the space of uniform images, and uniform images are generally not in the null space of $\boldsymbol{A}$. So Assumption (6) holds for most inverse problems. The above assumption also ensures that the objective function $\Phi$ is strictly convex (see (8)) when the $\psi_i$'s are twice differentiable with positive curvatures and satisfy the conditions above.

Even if the $\psi_i$'s are not twice differentiable, $\Phi$ is still convex due to the assumed convexity of the $\psi_i$'s. The argument showing $\Phi$ is convex is simple; for $\alpha \in [0,1]$:

$$\Phi(\alpha \underline{u} + (1-\alpha)\underline{v}) = \sum_{i=1}^m \psi_i([\boldsymbol{B}(\alpha \underline{u} + (1-\alpha)\underline{v}) - \underline{c}]_i) = \sum_{i=1}^m \psi_i(\alpha[\boldsymbol{B}\underline{u} - \underline{c}]_i + (1-\alpha)[\boldsymbol{B}\underline{v} - \underline{c}]_i)$$

$$\leq \sum_{i=1}^m \alpha \psi_i([\boldsymbol{B}\underline{u} - \underline{c}]_i) + (1-\alpha)\psi_i([\boldsymbol{B}\underline{v} - \underline{c}]_i) = \alpha\Phi(\underline{u}) + (1-\alpha)\Phi(\underline{v}).$$

We consider convexity to be desirable since non-convex objective functions lead to estimators that are discontinuous functions of the measurements [9]. Algorithms for the convex case are also of use for problems that are non-convex since graduated nonconvexity and deterministic annealing methods are often applied.

## III. HUBER'S ALGORITHM

In [4], Huber derived an algorithm for minimizing objective functions of the form (5) in the context of robust linear regression[1]. The algorithm can be considered both as a form of successive-substitution and as a form of iteratively reweighted least-squares. The ARTUR algorithm of [11] is essentially the special case of Huber's algorithm where $\psi_i^{\text{data}}(t) = t^2$.

We derive the algorithm by first identifying a recursive relationship for the stationary point(s) of the gradient of $\Phi$, which then suggests a successive-substitution algorithm. Throughout we assume that conditions C1 to C5 and Assumption (6) hold. We then discuss convergence.

### *A. Gradients and Convexity*

From (5) one can easily verify that (cf equation (8.31) of [4]):

$$\frac{\partial}{\partial x_j}\Phi(\underline{x}) = \sum_{i=1}^m b_{ij}\dot{\psi}_i([\boldsymbol{B}\underline{x} - \underline{c}]_i) = \sum_{i=1}^m b_{ij}\,\omega_i([\boldsymbol{B}\underline{x} - \underline{c}]_i)\,[\boldsymbol{B}\underline{x} - \underline{c}]_i$$

where $\dot{\psi}_i$ is the derivative of $\psi_i$ (which exists by condition C2) and where $\omega_i(t)$ is any function satisfying $t\omega_i(t) = \dot{\psi}_i(t)$ for all $t$ (which exists by condition C4). Thus in vector form:

$$\dot{\Phi}(\underline{x}) = \boldsymbol{B}'\boldsymbol{\Omega}(\underline{x})(\boldsymbol{B}\underline{x} - \underline{c}) = \boldsymbol{B}'\boldsymbol{\Omega}(\underline{x})\boldsymbol{B}\underline{x} - \boldsymbol{B}'\boldsymbol{\Omega}(\underline{x})\underline{c} \tag{7}$$

where $\dot{\Phi}$ denotes the column gradient of $\Phi$ and

$$\boldsymbol{\Omega}(\underline{x}) = \operatorname{diag}\{\omega_i([\boldsymbol{B}\underline{x} - \underline{c}]_i)\}$$

is a diagonal matrix of positive values (see condition P8). If the $\psi_i$'s are twice differentiable, then one can also easily verify that

$$\frac{\partial^2}{\partial x_j \partial x_k}\Phi(\underline{x}) = \sum_{i=1}^m b_{ij}b_{ik}\ddot{\psi}_i([\boldsymbol{B}\underline{x} - \underline{c}]_i).$$

[1]The extension of Huber's algorithm to the case where the $\psi_i$'s depend on $i$ is trivial.

Thus the Hessian of $\Phi$ is given by the expression

$$\nabla^2\Phi(\underline{x}) = \boldsymbol{B}'\boldsymbol{\Lambda}(\underline{x})\boldsymbol{B} \quad \text{where} \quad \boldsymbol{\Lambda}(\underline{x}) = \operatorname{diag}\left\{\ddot{\psi}_i([\boldsymbol{B}\underline{x} - \underline{c}]_i)\right\}. \tag{8}$$

It follows immediately from Assumption (6) that if the $\ddot{\psi}_i$'s are everywhere positive, then the Hessian of $\Phi$ is positive definite, and thus that $\Phi$ is a strictly convex function. Since the Huber function (3) is not twice differentiable and has zero curvature for $|t| > \delta$, it is desirable not to require these extra conditions.

### B. Successive Substitutions

If one disregards any hard constraints on $\underline{x}$ such as nonnegativity, then minimizers of $\Phi(\underline{x})$ must satisfy $\underline{0} = \dot{\Phi}(\hat{\underline{x}})$, so from (7):

$$[\boldsymbol{B}'\boldsymbol{\Omega}(\hat{\underline{x}})\boldsymbol{B}]\,\hat{\underline{x}} = \boldsymbol{B}'\boldsymbol{\Omega}(\hat{\underline{x}})\underline{c}.$$

This relationship is similar to the "normal equations" for linear least-squares estimation, except that iterative methods are required here because of the nonlinear dependence on $\hat{\underline{x}}$. The above equality suggests the following natural successive-substitutions iteration, which we refer to as Huber's algorithm:

$$\begin{aligned} \underline{x}^{n+1} &= [\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\boldsymbol{B}]^{-1}\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\underline{c} = \underline{x}^n - [\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\boldsymbol{B}]^{-1}\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)(\boldsymbol{B}\underline{x} - \underline{c}) \\ &= \underline{x}^n - [\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\boldsymbol{B}]^{-1}\dot{\Phi}(\underline{x}^n). \end{aligned} \tag{9}$$

(Note that $\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\boldsymbol{B}$ is invertible by (6).) Clearly if $\underline{x}^\infty$ is a fixed point of the iteration (9), then $\underline{x}^\infty$ satisfies $\underline{0} = \dot{\Phi}(\underline{x}^\infty)$. Since $\Phi$ is globally convex, any $\underline{x}$ for which $\dot{\Phi}(\underline{x})$ is zero is a minimizer of $\Phi$. Furthermore, when $\Phi$ is strictly convex its minimizer is unique, and thus $\underline{x}^\infty = \hat{\underline{x}}$. However, these facts alone do not ensure convergence.

### C. Reweighted Least-Squares Interpretation

If we define the following estimate-weighted least-squares objective function

$$\phi^{\mathrm{WLS}}(\underline{x}; \underline{x}^n) = \sum_{i=1}^m \omega_i([\boldsymbol{B}\underline{x}^n - \underline{c}]_i)\,\frac{1}{2}([\boldsymbol{B}\underline{x} - \underline{c}]_i)^2 = \frac{1}{2}(\boldsymbol{B}\underline{x} - \underline{c})'\boldsymbol{\Omega}(\underline{x}^n)(\boldsymbol{B}\underline{x} - \underline{c}), \tag{10}$$

then one can easily show that the successive substitutions iteration (9) can also be expressed as follows:

$$\underline{x}^{n+1} = \arg\min_{\underline{x}} \phi^{\mathrm{WLS}}(\underline{x}; \underline{x}^n). \tag{11}$$

This type of iteration is known as iteratively reweighted least-squares [12]. There is a subtle but important difference here though; in conventional reweighted least-squares iterations, the weights are usually the second derivatives of the $\psi_i$'s, which would lead to the iteration:

$$\underline{x}^{n+1} = \underline{x}^n - [\boldsymbol{B}'\boldsymbol{\Lambda}(\underline{x}^n)\boldsymbol{B}]^{-1}\dot{\Phi}(\underline{x}^n) = \underline{x}^n - [\nabla^2\Phi(\underline{x}^n)]^{-1}\dot{\Phi}(\underline{x}^n), \tag{12}$$

(cf (9)) where $\boldsymbol{\Lambda}(\underline{x})$ was defined in (8). The iteration (12) is identical to the Newton-Raphson iteration for minimizing $\Phi$. Unfortunately the Newton-Raphson iteration is not guaranteed to converge (or even to monotonically decrease $\Phi$) for the type of functions $\psi_i$ of interest here. Surprisingly, the small change of replacing the second derivatives in (12) with the $\omega_i$'s in (9) is enough of a difference to ensure monotonicity. Furthermore, Huber's iteration does not require the $\psi_i$'s to be twice differentiable! Note that Huber's iteration is identical to Newton-Raphson when all of the $\psi_i$'s are quadratic functions.

When the $\psi_i$'s are twice differentiable, then from P7 and P9 the second derivatives of each $\psi_i$ are bounded above by $\ddot{\psi}(0) = \omega_i(0)$. Thus one could make the Newton-Raphson iteration convergent by replacing $\boldsymbol{\Lambda}$ above with the diagonal matrix containing those upper bounds on the curvatures. However, this modified Newton-Raphson iteration will have a slower convergence rate than Huber's iteration, since by P8 the weights used in Huber's iteration have smaller values than the curvature upper bounds.

### D. Convergence Proof Sketch

In discussing the convergence of (9) or equivalently (11), Huber [4, p. 187] first showed that the iteration monotonically decreases the objective function, i.e. $\Phi(\underline{x}^{n+1}) \leq \Phi(\underline{x}^n)$, by showing that $\phi^{\mathrm{WLS}}$ is a "comparison function" that satisfies

$$\phi^{\mathrm{WLS}}(\underline{x}; \underline{x}^n) - \phi^{\mathrm{WLS}}(\underline{x}^n; \underline{x}^n) \geq \Phi(\underline{x}) - \Phi(\underline{x}^n). \tag{13}$$

This is precisely the property used to establish monotonicity of EM-type algorithms that can be written in the form (11) [13–15].

The monotonicity property ensures that the sequence $\{\Phi(\underline{x}^n)\}$ converges (since $\Phi$ is bounded below by $\sum_{i=1}^m \psi_i(0)$). However, Huber did not proceed to prove that the sequence of iterates $\{\underline{x}^n\}$ converges. Instead he established convergence of a somewhat different algorithm called the modified residuals method [4].

Using arguments very similar to those in [15, 16], we have shown that if $\Phi$ has a unique minimizer $\hat{\underline{x}}$, then $\{\underline{x}^n\}$ converges globally to $\hat{\underline{x}}$. (Strict convexity of $\Phi$ is sufficient but certainly not necessary to ensure $\Phi$ has a unique minimizer.) The details will be provided elsewhere, hopefully after resolving whether the sequence convergences to a minimizer even when there are multiple minima.

### E. Computational Complexity

Even when Huber's algorithm is globally convergent, it is impractical for imaging problems since it requires "inversion" of the matrix $\boldsymbol{B}'\boldsymbol{\Omega}(\underline{x}^n)\boldsymbol{B}$. Charbonnier *et al.* addressed this problem by applying iterative algorithms such as conjugate-gradient and Gauss-Siedel iterations to find an approximate minimum of the quadratic form in (11) [11]. Such use of iterative linear-equation solvers as subiterations within the main iterations is fairly expensive computationally, and may therefore be suboptimal for imaging problems. Furthermore the Huber iteration does not easily accommodate nonnegativity constraints for $\underline{x}$, which are often important in imaging problems. An additional minor consideration is that the use of approximate solutions to (11) raises questions about the guarantee of convergence. (The convergence proofs in [11] assume that (11) is exactly minimized each iteration, even though this is not achievable in practice.) Therefore, for imaging problems it is desirable to find new algorithms that are globally convergent but that require less expensive subiterations.

## IV. GROUPED COORDINATE DESCENT ALGORITHM

In [8] we presented a new algorithm for penalized-likelihood tomographic image reconstruction from Poisson transmission measurements. The basic principles of the algorithm apply much more broadly than to the specific problem addressed in [8]. Here we use those principles to derive a general algorithm suitable for finding a minimizer of objective functions of the form (5).

The basic idea is that since minimizing $\Phi$ over all elements of $\underline{x}$ is a difficult problem, we instead select a subset of the pixels and first (partially) minimize $\Phi$ over those pixels and then move onto another group of pixels. Let $\mathcal{S}$ be a subset of the pixels $\{1, \ldots, p\}$, let $\tilde{\mathcal{S}}$ be its complement, and let $|\mathcal{S}|$ be the cardinality of $\mathcal{S}$. In a grouped-coordinate descent (GCD) algorithm[2], we update $\underline{x}_{\mathcal{S}}$ while holding $\underline{x}^n_{\tilde{\mathcal{S}}}$ fixed at the $n$th update [8, 14]. Unfortunately it is even too difficult to minimize $\Phi(\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}})$ over $\underline{x}_{\mathcal{S}}$ directly, so we will settle for finding a method for choosing $\underline{x}^{n+1}_{\mathcal{S}}$ that will at least *monotonically decrease* the objective function[3]:

$$\Phi(\underline{x}^{n+1}) = \Phi(\underline{x}^{n+1}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}) \leq \Phi(\underline{x}^n_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}) = \Phi(\underline{x}^n).$$

Instead of trying to find $\underline{x}^{n+1}_{\mathcal{S}}$ to minimize $\Phi(\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}})$, we minimize a *surrogate function* $\phi(\underline{x}_{\mathcal{S}}; \underline{x}^n)$ that we must choose to satisfy the condition (13). The GCD update (cf SAGE algorithm [14, 15]) is then:

$$\begin{aligned} \underline{x}^{n+1}_{\mathcal{S}} &= \arg\min_{\underline{x}_{\mathcal{S}}} \phi(\underline{x}_{\mathcal{S}}; \underline{x}^n), \\ x^{n+1}_j &= x^n_j, \quad j \in \tilde{\mathcal{S}}. \end{aligned} \tag{14}$$

[2] In a GCD method, $\mathcal{S}$ varies with $n$. To simplify notation, we leave this dependence implicit. Also note the notation abuse: $\Phi(\underline{x}_{\mathcal{S}}, \underline{x}_{\tilde{\mathcal{S}}}) = \Phi(\underline{x})$.

[3] To simplify notation, in the presentation we increment $n$ every time a group of pixels is updated. We reserve the term "iteration" to mean a complete update of all pixels.

The condition (13) is sufficient to ensure that the iterates produced by the above generic update will monotonically decrease the objective: $\Phi(\underline{x}^{n+1}) \leq \Phi(\underline{x}^n)$.

*A. Choosing Surrogate Functions*

We restrict attention here to additively separable[4] surrogate functions $\phi(\cdot;\underline{x}^n)$ satisfying

$$\phi(\underline{x}_{\mathcal{S}};\underline{x}^n) = \sum_{j\in\mathcal{S}} \phi_j(x_j;\underline{x}^n). \tag{15}$$

To choose these $\phi_j$'s, we use modifications of De Pierro's convexity method [17, 18]. The key step is to note that if $\underline{r}(\underline{x}) = \boldsymbol{B}\underline{x} - \underline{c}$, then

$$r_i([\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}]) = \sum_{j\in\mathcal{S}} \alpha_{ij} \left[\frac{b_{ij}}{\alpha_{ij}}(x_j - x_j^n) + r_i(\underline{x}^n)\right] \tag{16}$$

for any choice[5] of $\alpha_{ij} \geq 0$ that satisfies the constraint

$$\sum_{j\in\mathcal{S}} \alpha_{ij} = 1, \quad \forall i. \tag{17}$$

We discussed specific choices for $\alpha_{ij}$ in [8]. One particularly useful choice is

$$\alpha_{ij} = |b_{ij}| / \sum_{k\in\mathcal{S}} |b_{ik}|. \tag{18}$$

Since $\psi_i$ is convex over all of $\mathbb{R}$ by P3, it follows directly from (16) that:

$$\psi_i(r_i([\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}])) \leq \sum_{j\in\mathcal{S}} \alpha_{ij}\, \psi_i\left(\frac{b_{ij}}{\alpha_{ij}}(x_j - x_j^n) + r_i(\underline{x}^n)\right). \tag{19}$$

It follows from (19) that we have:

$$\Phi([\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}]) = \sum_{i=1}^{m} \psi_i(r_i([\underline{x}_{\mathcal{S}}, \underline{x}^n_{\tilde{\mathcal{S}}}])) \leq \sum_{j\in\mathcal{S}} \phi_j(x_j;\underline{x}^n)$$

for all $\underline{x}_{\mathcal{S}}$, where using (16) and (19):

$$\phi_j(x;\underline{x}^n) = \sum_{i\in\mathcal{I}_j} \alpha_{ij}\, \psi_i\left(\frac{b_{ij}}{\alpha_{ij}}(x - x_j^n) + r_i(\underline{x}^n)\right), \tag{20}$$

and where

$$\mathcal{I}_j = \{i : b_{ij} \neq 0\}.$$

The surrogate function defined by (15) with (20) will satisfy the monotonicity condition (13). Each $\phi_j$ only depends on one $x_j$, so the minimization step in (14) reduces to $|\mathcal{S}|$ separate 1D minimizations. Thus (14) becomes the parallelizable operations:

$$x_j^{n+1} = \arg\min_x \phi_j(x;\underline{x}^n), \quad j \in \mathcal{S}. \tag{21}$$

Note that because the minimization is separable, it is trivial to incorporate nonnegativity constraints [8]. Borrowing from the expectation-maximization algorithm [13, 14], we refer to these minimizations as the "M-step" of the algorithm. (Computing $r_i(\underline{x}^n)$ can be considered the "E-step.")

[4] Separable surrogate functions are very convenient for enforcing the nonnegativity constraint. There may be alternatives that lead to faster convergence though.
[5] We assume $\alpha_{ij} = 0$ if and only if $b_{ij} = 0$ so that (16) is well defined, which is satisfied by the choice (18).

### *B. The Minimization Step*

The surrogate functions $\phi_j$ defined by (20) for fixed $\underline{x}^n$ are themselves a special case of the general form (5). Thus we can apply the successive-substitutions method described by (9) in Section 3 to the minimization required in (21). This new approach is a significant improvement over the M-step methods described in [8].

To simplify notation, consider the update of a particular pixel $j$ for a particular iteration $n$, so that we can drop the dependence on $j$ and $n$. Specifically, define

$$\phi(x) = \phi_j(x; \underline{x}^n) \quad \text{and} \quad \tilde{\psi}_i(t) = \alpha_{ij} \psi_i\left(\frac{b_{ij}}{\alpha_{ij}} t\right)$$

so that from (20):

$$\phi(x) = \sum_{i \in \mathcal{I}_j} \tilde{\psi}_i(x - z_i) \tag{22}$$

where

$$z_i = -x_j^n + \frac{\alpha_{ij}}{b_{ij}} r_i(\underline{x}^n).$$

The natural sub-iteration for updating $x_j$ is thus to initialize with $x^{\text{old}} = x_j^n$, and then iterate

$$x^{\text{new}} = M(x^{\text{old}}), \tag{23}$$

where, applying (9) to (22), the mapping for the Huber iteration is:

$$M(x) = x - \frac{\dot{\phi}(x)}{\sum_{i \in \mathcal{I}_j} \omega_{\tilde{\psi}_i}(x - z_i)}. \tag{24}$$

After a few sub-iterations of this mapping, we assign $x_j^{n+1}$ to the current value of $x^{\text{new}}$.

The overall algorithm is summarized in Table I, where $[x]_+ = x$ for $x > 0$ and is 0 otherwise. This $[\cdot]_+$ operator enforces the nonnegativity constraint, which is optional. The symbol ":=" denotes assignment.

### *C. Convergence*

We have extended the proof given in [15] to establish convergence of the grouped coordinate descent algorithm described above, provided that the minimizer of $\Phi$ is unique and that the groups $\mathcal{S}$ are chosen such that every pixel is updated periodically, as discussed in [15]. The extension of the proof accommodates the fact that the surrogate functions $\phi_j(\cdot; \underline{x}^n)$ are not exactly minimized by a finite number of subiterations of the form (24). The proofs also include the case where the nonnegativity constraint is included by modifying the mapping to be $x^{\text{new}} = \max\{M(x^{\text{old}}), 0\}$. Details will be provided elsewhere; interested readers can check the author's web site.

It is of interest to specifically examine the convergence behavior of the 1-D subiteration (24). The next section analyzes this iteration.

## V. THE 1-D SUBITERATION

A key component of several algorithms for image reconstruction and restoration (often part of the so-called M-step) is the minimization of a scalar functional of the form

$$\phi(x) = \sum_{i=1}^{m} \psi_i(x - z_i).$$

Examples include [8, 19, 20] and (22) above. Ignoring any nonnegativity constraint on $x$ (although this constraint could be included in the analysis), we would like to minimize $\phi(x)$ by zeroing its derivative:

$$\dot{\phi}(x) = \sum_{i=1}^{m} \dot{\psi}_i(x - z_i).$$

Initialize $\underline{\hat{x}}$. Precompute $\underline{r} := \boldsymbol{B}\underline{\hat{x}} - \underline{c}$.

```
for each iteration:
    for each S:
        for each j ∈ S:
```

$$x_j^{\text{work}} := \hat{x}_j$$

```
            for a couple subiterations:
```

$$x_j^{\text{work}} := \left[ x_j^{\text{work}} - \frac{\sum_{i=1}^m b_{ij} \, \dot{\psi}\left( \frac{b_{ij}}{\alpha_{ij}} (x_j^{\text{work}} - \hat{x}_j) + r_i \right)}{\sum_{i=1}^m \frac{b_{ij}^2}{\alpha_{ij}} \, \omega_i \left( \frac{b_{ij}}{\alpha_{ij}} (x_j^{\text{work}} - \hat{x}_j) + r_i \right)} \right]_+ \tag{25}$$

```
            end
```

$$r_i := r_i + b_{ij} \, (x_j^{\text{work}} - \hat{x}_j), \quad \forall i \text{ s.t. } b_{ij} \neq 0 \tag{26}$$

$$\hat{x}_j := x_j^{\text{work}}$$

```
        end
    end
end
```

TABLE I

THE GENERAL GROUPED COORDINATE DESCENT ALGORITHM. NOTE THAT THE UPDATES OF $\hat{x}_j$ ARE DONE "IN PLACE."

Here we are specifically interested in a "modified" Newton-Raphson iteration of the form $x^{\text{new}} = M(x^{\text{old}})$, where $M(\cdot)$ has the following form:

$$M(x) = x - \frac{\dot{\phi}(x)}{d(x)} = x - \frac{\dot{\phi}(x)}{\sum_{i=1}^m \omega_i(x - z_i)} = x - \frac{\sum_{i=1}^m \dot{\psi}_i(x - z_i)}{\sum_{i=1}^m \omega_i(x - z_i)}. \tag{27}$$

The standard Newton-Raphson method uses $\omega_i(t) = \ddot{\psi}_i(t)$, but this choice does not guarantee convergence in general. One could ensure convergence by using $\omega_i(t) = \max_t \ddot{\psi}_i(t) = \ddot{\psi}(0)$ (see P9 and P7) when $\psi$ is twice differentiable with bounded curvature. But this choice leads to very slow convergence rates. Instead, we focus here on the special case where $\omega_i(t) = \dot{\psi}_i(t)/t$, as defined by C4. That choice has its origins in Huber's iteration, as well as in the "half-quadratic" method for nonquadratic penalty functions [11, 21, 22].

Since $\phi$ is convex but not necessarily strictly convex, the set of minimizers is either a single point or an interval of finite length. (Conditions C1 to C5 do not ensure that the minimizer of $\phi$ is unique.) From (27), the set of minimizers of $\phi$ equals the set of fixed points of $M$, i.e., where $M(\hat{x}) = \hat{x}$. Huber's convergence theorems [4] established that the mapping $M(\cdot)$ monotonically decreases $\phi$, but did not establish convergence to a fixed point of $M(\cdot)$.

### *A. Nonexpansive*

The following result shows that the iteration $M$ is nonexpansive, i.e., the sequence of iterates "gets closer" to a minimizer each iteration. This is a somewhat stronger result than the fact that $M$ provides a sequence of iterates for which $\phi$ is monotonically non-increasing.

**Theorem.**
If the $\psi_i$'s satisfy the conditions C1 to C5, then for any $z_i$'s $M(\cdot)$ is *globally nonexpansive*, i.e. $|M(x) - \hat{x}| \leq |x - \hat{x}|$ where $\hat{x}$ is any fixed-point of $M(\cdot)$, for any $x \neq \hat{x}$.

*Proof:*

First note that

$$\frac{\dot\phi(x)-\dot\phi(\hat x)}{x-\hat x} = \frac{\sum_{i=1}^m \dot\psi_i(x-z_i)-\dot\psi_i(\hat x - z_i)}{x-\hat x} = \sum_{i=1}^m \frac{\dot\psi_i(x-z_i)-\dot\psi_i(\hat x-z_i)}{x-z_i-(\hat x - z_i)} = \sum_{i=1}^m \lambda_{\psi_i}(x-z_i, \hat x - z_i)$$

where

$$\lambda_\psi(u,v) = \begin{cases} \dfrac{\dot\psi(u)-\dot\psi(v)}{u-v}, & u \neq v \\ \omega_\psi(u), & u = v \end{cases}. \tag{28}$$

Thus for $x \neq \hat x$:

$$\frac{|M(x)-\hat x|}{|x-\hat x|} = \frac{\left| x - \frac{\dot\phi(x)}{d(x)} - \hat x \right|}{|x-\hat x|} = \frac{\left| x - \hat x - \frac{\dot\phi(x)-\dot\phi(\hat x)}{d(x)} \right|}{|x-\hat x|} = \left| 1 - \frac{\dot\phi(x)-\dot\phi(\hat x)}{d(x)(x-\hat x)} \right|$$

$$= \left| 1 - \frac{\sum_{i=1}^m \lambda_{\psi_i}(x-z_i, \hat x - z_i)}{\sum_{i=1}^m \omega_i(x-z_i)} \right| = \frac{\left|\sum_{i=1}^m \omega_i(x-z_i)[1-\lambda_{\psi_i}(x-z_i,\hat x - z_i)/\omega_i(x-z_i)]\right|}{\sum_{i=1}^m \omega_i(x-z_i)}. \tag{29}$$

Thus since $\omega_i(x-z_i) > 0$, for $M(\cdot)$ to be nonexpansive it is *sufficient* to have

$$|1 - \lambda_\psi(u,v)/\omega_\psi(u)| \leq 1 \tag{30}$$

for all $u \neq v$ and for all $\psi$ satisfying C1 to C5. This condition is established by the following lemma. □

We conjecture that one can also show that $M$ is a contraction, i.e., $|M(x)-\hat x| < |M(x) - \hat x|$ provided $M(x) \neq x$, i.e. $x$ is not a minimizer of $\phi$.

*Lemma 1:* For $\psi$ satisfying conditions C1 to C5:

$$0 \leq \frac{\lambda(u,v)}{\omega(u)} < 2,$$

where $\lambda$ was defined by (28) and for notational simplicity we drop the dependence on $\psi$ here.

Proof:

For $u = v$, both equalities are clearly satisfied, since $\lambda(u,u)/\omega(u) = 1$. For $u \neq v$, the left-hand inequality follows immediately from C3. We now focus on the right-hand equality in the case $u \neq v$.

Case I: $v = 0 \neq u$. From (28) it is clear that $\lambda(u,0)/\omega(u) = 1$.

Case II: $u = 0 \neq v$. $\lambda(0,v)/\omega(0) = \omega(v)/\omega(0) \leq 1$ by P8.

Case III: $v = -u \neq 0$. $\lambda(u,-u)/\omega(u) = 1$.

Case IV: $0 < |v| < |u|$. In this case, establishing the right-hand inequality is equivalent to showing

$$\lambda(u,v) = \frac{\omega(u) - \dot\psi(v)/u}{1 - v/u} < 2\omega(u)$$

or equivalently that

$$\omega(u) - \dot\psi(v)/u < 2\omega(u)(1 - v/u)$$

since $|v| < |u|$ implies $1 - v/u$ is positive. Dividing through by $\omega(u)$ and simplifying, we must show

$$2\frac{v}{u} < 1 + \frac{\dot\psi(v)}{\dot\psi(u)}. \tag{31}$$

If $u$ and $v$ have the same sign, then by first using the fact that $|v|/|u| < 1$ in this case, and then that $\omega(u) \leq \omega(v)$ (by C4 and P6), we have

$$2\frac{v}{u} = 2\left|\frac{v}{u}\right| < 1 + \left|\frac{v}{u}\right| \leq 1 + \left|\frac{v}{u}\right| \frac{\omega(v)}{\omega(u)} = 1 + \left|\frac{\dot\psi(v)}{\dot\psi(u)}\right| = 1 + \frac{\dot\psi(v)}{\dot\psi(u)}$$

which confirms (31). Now if $u$ and $v$ have opposite signs, then $2v/u < 0$, but $1+\dot{\psi}(v)/\dot{\psi}(u) \geq 0$ since $|\dot{\psi}(v)/\dot{\psi}(u)| \leq 1$ by P5 for $|v| < |u|$. Thus (31) holds regardless of the signs of $u$ and $v$.

Case V: $0 < |u| < |v|$. Using the symmetry of $\lambda(u, v)$ for $u \neq v$, and since $\omega(v) \leq \omega(u)$ for $|u| < |v|$ by C4 and P6, $\lambda(u, v)/\omega(u) = \lambda(v, u)/\omega(u) \leq \lambda(v, u)/\omega(v) < 2$ by Case IV.

Thus we have shown $0 \leq \lambda(u, v)/\omega(u) < 2$ for any $u, v$ pair. □

## VI. ACKNOWLEDGEMENT

This work was supported in part by NIH grants CA-60711 and CA-54362 and the Whitaker Foundation.

## References

[1] A H Delaney and Y Bresler. A fast and accurate Fourier algorithm for iterative parallel-beam tomography. *IEEE Tr. Im. Proc.*, 5(5):740–53, May 1996.
[2] Gabor T Herman. *Image reconstruction from projections: The fundamentals of computerized tomography.* Academic Press, New York, 1980.
[3] R M Lewitt. Multidimensional digital image representations using generalized Kaiser-Bessel window functions. *J. Opt. Soc. Amer. Ser. A*, 7(10):1834–1846, October 1990.
[4] P J Huber. *Robust statistics.* Wiley, New York, 1981.
[5] J A Fessler. Penalized weighted least-squares image reconstruction for positron emission tomography. *IEEE Tr. Med. Im.*, 13(2):290–300, June 1994.
[6] D L Snyder, C W Helstrom, A D Lanterman, M Faisal, and R L White. Compensation for readout noise in CCD images. *J. Opt. Soc. Amer. Ser. A*, 12(2):272–83, February 1995.
[7] J A Fessler and W L Rogers. Spatial resolution properties of penalized-likelihood image reconstruction methods: Space-invariant tomographs. *IEEE Tr. Im. Proc.*, 5(9):1346–58, September 1996.
[8] J A Fessler, E P Ficaro, N H Clinthorne, and K Lange. Grouped-coordinate ascent algorithms for penalized-likelihood transmission image reconstruction. *IEEE Tr. Med. Im.*, 16(2):166–75, April 1997.
[9] C Bouman and K Sauer. A generalized Gaussian image model for edge-preserving MAP estimation. *IEEE Tr. Im. Proc.*, 2(3):296–310, July 1993.
[10] H L Royden. *Real analysis.* Macmillan, New York, 3 edition, 1988.
[11] P Charbonnier, L Blanc-Féraud, G Aubert, and M Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Tr. Im. Proc.*, 6(2), February 1997.
[12] P J Green. Iteratively reweighted least squares for maximum likelihood estimation, and some robust and resistant alternatives. *J. Royal Stat. Soc. Ser. B*, 46(2):149–192, 1984.
[13] A P Dempster, N M Laird, and D B Rubin. Maximum likelihood from incomplete data via the EM algorithm. *J. Royal Stat. Soc. Ser. B*, 39(1):1–38, 1977.
[14] J A Fessler and A O Hero. Space-alternating generalized expectation-maximization algorithm. *IEEE Tr. Sig. Proc.*, 42(10):2664–77, October 1994.
[15] J A Fessler and A O Hero. Penalized maximum-likelihood image reconstruction using space-alternating generalized EM algorithms. *IEEE Tr. Im. Proc.*, 4(10):1417–29, October 1995.
[16] K Lange and R Carson. EM reconstruction algorithms for emission and transmission tomography. *J. Comp. Assisted Tomo.*, 8(2):306–316, April 1984.
[17] A R De Pierro. On the relation between the ISRA and the EM algorithm for positron emission tomography. *IEEE Tr. Med. Im.*, 12(2):328–333, June 1993.
[18] A R De Pierro. A modified expectation maximization algorithm for penalized likelihood estimation in emission tomography. *IEEE Tr. Med. Im.*, 14(1):132–137, March 1995.
[19] C A Bouman and K Sauer. A unified approach to statistical tomography using coordinate descent optimization. *IEEE Tr. Im. Proc.*, 5(3):480–92, March 1996.
[20] J A Fessler and S D Booth. Conjugate-gradient preconditioning methods for shift-variant image reconstruction. *IEEE Tr. Im. Proc.*, 1997. Submitted.
[21] D Geman and G Reynolds. Constrained restoration and the recovery of discontinuities. *IEEE Tr. Patt. Anal. Mach. Int.*, 14(3):367–383, March 1992.
[22] D Geman and C Yang. Nonlinear image recovery with half-quadratic regularization. *IEEE Tr. Im. Proc.*, 4(7):932–46, July 1995.