IPEM
Institute of Physics and Engineering in Medicine

PAPER

Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle–Pock algorithm

View the article online for updates and enhancements.

Convex optimization problem prototyping for image reconstruction in computed tomography with the Chambolle–Pock algorithm

Emil Y Sidky[1]**, Jakob H Jørgensen**[2] **and Xiaochuan Pan**[1]

[1] Department of Radiology, University of Chicago, 5841 S. Maryland Ave., Chicago, IL 60637, USA
[2] Department of Informatics and Mathematical Modeling, Technical University of Denmark, Richard Petersens Plads, Building 321, 2800 Kgs. Lyngby, Denmark

E-mail: sidky@uchicago.edu, jakj@imm.dtu.dk and xpan@uchicago.edu

Abstract

The primal–dual optimization algorithm developed in Chambolle and Pock (CP) (2011 *J. Math. Imag. Vis.* **40** 1–26) is applied to various convex optimization problems of interest in computed tomography (CT) image reconstruction. This algorithm allows for rapid prototyping of optimization problems for the purpose of designing iterative image reconstruction algorithms for CT. The primal–dual algorithm is briefly summarized in this paper, and its potential for prototyping is demonstrated by explicitly deriving CP algorithm instances for many optimization problems relevant to CT. An example application modeling breast CT with low-intensity x-ray illumination is presented.

(Some figures may appear in colour only in the online journal)

1. Introduction

Optimization-based image reconstruction algorithms for CT have been investigated heavily recently due to their potential to allow for reduced scanning effort while maintaining or improving image quality (McCollough *et al* 2009, Pan *et al* 2009). Such methods have been considered for many years, but during the past five years computational barriers have been lowered enough such that iterative image reconstruction can be considered for practical application in CT (Ziegler *et al* 2008). The transition to practice has been taking place alongside further theoretical development particularly with algorithms based on the sparsity-motivated ℓ_1-norm (Li *et al* 2002, Sidky *et al* 2006, 2010, Sidky and Pan 2008, Chen *et al* 2008, Ritschl *et al* 2011, Defrise *et al* 2011, Ramani and Fessler 2011, Jørgensen *et al* 2011a). Despite the recent interest in sparsity, optimization-based image reconstruction algorithm development continues to proceed along many fronts and there is as of yet no consensus on a particular

optimization problem for the CT system. In fact, it is beginning to look like the optimization problem, upon which the iterative image reconstruction algorithms are based, will themselves be subject to design depending on the particular properties of each scanner type and imaging task.

Considering the possibility of tailoring optimization problems to a class of CT scanners makes the design of iterative image reconstruction algorithms a daunting task. Optimization formulations generally construct an objective function comprised of a data fidelity term and possible penalty terms discouraging unphysical behavior in the reconstructed image, and they possibly include hard constraints on the image. The image estimate is arrived at by extremizing the objective subject to any constraints placed on the estimate. The optimization problems for image reconstruction can take many forms depending on image representation, projection model and objective and constraint design. On top of this, it is difficult to solve many of the optimization problems of interest. A change in optimization problem formulation can mean many weeks or months of algorithm development to account for the modification.

Due to this complexity, it would be quite desirable to have an algorithmic tool to facilitate design of optimization problems for CT image reconstruction. This tool would consist of a well-defined set of mechanical steps that generate a convergent algorithm from a specific optimization problem for CT image reconstruction. The goal of this tool would be to allow for rapid prototyping of various optimization formulations; one could design the optimization problem free of any restrictions imposed by a lack of an algorithm to solve it. The resulting algorithm might not be the most efficient solver for the particular optimization problem, but it would be guaranteed to give the answer.

In this paper we consider convex optimization problems for CT image reconstruction, including non-smooth objectives, unconstrained and constrained formulations. One general algorithmic tool is to use steepest descent or projected steepest descent (Nocedal and Wright 2006). Such algorithms, however, do not address non-smooth objective functions and they have difficulty with constrained optimization, being applicable for only simple constraints such as non-negativity. Another general strategy involves some form of evolving quadratic approximation to the objective. The literature on this flavor of algorithm design is enormous, including nonlinear conjugate gradient (CG) methods (Nocedal and Wright 2006), parabolic surrogates (Erdogan and Fessler 1999, Defrise *et al* 2011) and iteratively reweighted least-squares (Green 1984). For the CT system, these strategies often require quite a bit of know-how due to the very large scale and ill-posedness of the imaging model. Once the optimization formulation is established, however, these quadratic methods provide a good option to gain in efficiency.

One of the main barriers to prototyping alternative optimization problems for CT image reconstruction is the size of the imaging model; volumes can contain millions of voxels and the sinogram data can correspondingly consist of millions of x-ray transmission measurements. For large-scale systems, there has been some resurgence of first-order methods (Yin *et al* 2008, Combettes and Pesquet 2008, Beck and Teboulle 2009, Becker *et al* 2010, Chambolle and Pock 2011, Jensen *et al* 2011) and recently there has been applications of first-order methods specifically for optimization-based image reconstruction in CT (Jensen *et al* 2011, Choi *et al* 2010, Jørgensen *et al* 2011b). These methods are interesting because they can be adapted to a wide range of optimization problems involving non-smooth functions such as those involving ℓ_1-based norms. In particular, the algorithm that we pursue further in this paper is a first-order primal–dual algorithm for convex problems by Chambolle and Pock (2011). This algorithm goes a long way toward the goal of optimization problem prototyping because it covers a very general class of optimization problems that contain many optimization formulations of interest to the CT community.

For a selection of optimization problems of relevance to CT image reconstruction, we work through the details of setting up the Chambolle–Pock (CP) algorithm. We refer to these dedicated algorithms as *algorithm instances*. Our numerical results demonstrate that the algorithm instances achieve the solution of difficult convex optimization problems under challenging conditions in reasonable time and without parameter tuning. In section 2, the CP methodology and algorithm is summarized; in section 3, various optimization problems for CT image reconstruction are presented along with their corresponding CP algorithm instance and section 4 shows a limited study on a breast CT simulation that demonstrates the application of the derived CP algorithm instances.

2. Summary of the generic CP algorithm

The Chambolle and Pock (2011) (CP) algorithm is primal–dual meaning that it solves an optimization problem simultaneously with its dual. On its face, it would seem to involve extra work by solving two problems instead of one, but the algorithm comes with convergence guarantee and solving both problems provides a robust, non-heuristic convergence check—the duality gap.

The CP algorithm applies to a general form of the primal minimization:

$$\min_x \{F(Kx) + G(x)\}, \tag{1}$$

and a dual maximization:

$$\max_y \{-F^*(y) - G^*(-K^T y)\}, \tag{2}$$

where x and y are finite-dimensional vectors in the spaces X and Y, respectively; K is a linear transform from X to Y; G and F are convex, possibly non-smooth, functions mapping the respective X and Y spaces to non-negative real numbers and the superscript '*' in the dual maximization problem refers to convex conjugation, defined in equations (3) and (4). We note that the matrix K need not be a square; X and Y will in general have different dimensions. Given a convex function H of a vector $z \in Z$, its conjugate can be computed by the Legendre transform (Rockafellar *et al* 1970), and the original function can be recovered by applying conjugation again:

$$H^*(z) = \max_{z'} \{\langle z, z' \rangle_Z - H(z')\}, \tag{3}$$

$$H(z') = \max_z \{\langle z', z \rangle_Z - H^*(z)\}. \tag{4}$$

The notation $\langle \cdot, \cdot \rangle_Z$ refers to the inner product in the vector space Z.

Formally, the primal and dual problems are connected in a generic saddle point optimization problem:

$$\min_x \max_y \{\langle Kx, y \rangle_Y + G(x) - F^*(y)\}. \tag{5}$$

By performing the maximization over y in equation (5), using equation (4) with Kx associated with y', the primal minimization (1) is derived. Similarly, performing the minimization over x in equation (5), using equation (3) and the identity $\langle Kx, y \rangle = \langle x, K^T y \rangle$, yields the dual maximization (2), where the T superscript denotes matrix transposition.

The minimization problem in equation (1), though compact, covers many minimization problems of interest to tomographic image reconstruction. Solving the dual problem, equation (2), simultaneously allows for the assessment of algorithm convergence. For intermediate estimates x and y of the primal minimization and the dual maximization, respectively, the primal objective will be greater than or equal to the dual objective. The

difference between these objectives is referred to as the duality gap, and convergence is achieved when this gap is zero. Plenty of examples of useful optimization problems for tomographic image reconstruction will be described in detail in section 3, but first we summarize algorithm 1 from Chambolle and Pock (2011).

Algorithm 1. Pseudocode for N steps of the basic Chambolle–Pock algorithm. The constant L is the ℓ_2-norm of the matrix K; τ and σ are non-negative CP algorithm parameters, which are both set to $1/L$ in the present application; $\theta \in [0, 1]$ is another CP algorithm parameter, which is set to 1; and n is the iteration index. The proximal operators $prox_\sigma$ and $prox_\tau$ are defined in equation (6).

1: $L \leftarrow \|K\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize x_0 and y_0 to zero values
3: $\bar{x}_0 \leftarrow x_0$
4: **repeat**
5: $\quad y_{n+1} \leftarrow prox_\sigma[F^*](y_n + \sigma K\bar{x}_n)$
6: $\quad x_{n+1} \leftarrow prox_\tau[G](x_n - \tau K^T y_{n+1})$
7: $\quad \bar{x}_{n+1} \leftarrow x_{n+1} + \theta(x_{n+1} - x_n)$
8: $\quad n \leftarrow n + 1$
9: **until** $n \geqslant N$

2.1. Chambolle–Pock: algorithm 1

The CP algorithm simultaneously solves equations (1) and (2). As presented in Chambolle and Pock (2011), the algorithm is simple, yet extremely effective. We repeat the steps here in listing 1 for completeness, providing the parameters that we use for all results shown below. The parameter descriptions are provided in Chambolle and Pock (2011), but note that in our usage specified above there are *no* free parameters. This is an extremely important feature for our purpose of optimization prototyping. One caveat is that technically the proof of convergence for the CP algorithm assumes $L^2\sigma\tau < 1$, but in practice we have never encountered a case where the choice $\sigma = \tau = 1/L$ failed to tend to convergence. We stress that in equation (2) the matrix K^T needs to be the transpose of the matrix K; this point can sometimes be confusing because K for imaging applications is often intended to be an approximation to some continuous operator such as projection or differentiation and often K^T is taken to mean the approximation to the continuous operator's adjoint, which may or may not be the matrix transpose of K. The constant L is the magnitude of the matrix K, its largest singular value. Appendix A gives the details on computing L via the power method. The key to deriving the particular algorithm instances are the proximal mappings $prox_\sigma[F^*]$ and $prox_\tau[G]$ (called resolvent operators in Chambolle and Pock (2011)).

The proximal mapping is used to generate a descent direction for the convex function H and it is obtained by the following minimization:

$$prox_\sigma[H](z) = \arg\min_{z'} \left\{ H(z') + \frac{\|z - z'\|_2^2}{2\sigma} \right\}. \tag{6}$$

This operation does admit non-smooth convex functions, but H does need to be simple enough that the above minimization can be solved in a closed form. For CT applications, the ability to handle non-smooth F and G allows the study of many optimization problems of recent interest, and the simplicity limitation is not that restrictive as will be seen.

2.2. The CP algorithm for prototyping of convex optimization problems

To prototype a particular convex optimization problem for CT image reconstruction with the CP algorithm, there are five basic steps.

(1) Map the optimization problem to the generic minimization problem in equation (1).

(2) Derive the dual maximization problem, equation (2), by computing the convex conjugates of F and G using the Legendre transform (3).

(3) Derive the proximal mappings of F^* and G using equation (6).

(4) Substitute the results of (3) into the generic CP algorithm in listing 1 to obtain a CP algorithm instance.

(5) Run the algorithm, monitoring the primal–dual gap for convergence.

As will be seen below, a great variety of constrained and unconstrained optimization problems can be written in the form of equation (1). Specifically, using the algebra of convex functions (Rockafellar *et al* 1970), that the sum of two convex functions is convex and that the composition of a convex function with a linear transform is a convex function, many interesting optimization formulations can be put in the form of equation (1). We will also make use of convex functions which are not smooth—notably ℓ_1 based norms and indicator functions $\delta_S(x)$:

$$\delta_S(x) \equiv \begin{cases} 0 & x \in S \\ \infty & x \notin S \end{cases},\tag{7}$$

where S is a convex set. The indicator function is particularly handy for imposing constraints. In computing the convex conjugate and proximal mapping of convex functions, we make much use of the standard calculus rule for extremization, $\nabla f = 0$, but such computations are also augmented with geometric reasoning, which may be unfamiliar. Accordingly, we have included appendices to show some of these computation steps. With this quick introduction, we are now in a position to derive various algorithm instances for CT image reconstruction from different convex optimization problems.

3. CP algorithm instances for CT

For this paper, we only consider optimization problems involving the linear imaging model for x-ray projection, where the data are considered as line integrals over the object's x-ray attenuation coefficient. Generically, maintaining consistent notation with Chambolle and Pock (2011), the discrete-to-discrete CT system model (Barrett and Myers 2004) can be written as

$$Au = g,\tag{8}$$

where A is the projection matrix taking an object represented by expansion coefficients u and generating a set of line-integration values g. This model covers a multitude of expansion functions and CT configurations, including both 2D fan-beam and 3D cone-beam projection data models.

A few notes on notation are in order. In the following, we largely avoid indexing of the various vector spaces in order that the equations and pseudocode listings are brief and clear. Any of the standard algebraic operations between vectors is to be interpreted in a componentwise manner unless explicitly stated. Also an algebraic operation between a scalar and a vector is to be distributed among all components of the vector, e.g. $1 + v$ adds one to all components of v. For the optimization problems below, we employ three vector spaces: I the space of discrete images in either 2 or 3 dimensions; D the space of the CT sinograms (or projection data); and V the space of spatial-vector-valued image arrays, $V = I^d$, where $d = 2$ or 3 for 2D and 3D-space, respectively. For the CT system model (equation (8)), $u \in I$

and $g \in D$, but we note that the space D can also include sinograms which are not consistent with the linear system matrix A. The vector space V will be used below for forming the total variation (TV) semi-norm; an example of such a vector $v \in V$ is the spatial gradient of an image u. Although the pixel representation is used, much of the following can be applied to other image expansion functions. As we will be making much use of certain indicator functions, we define two important sets, $Box(a)$ and $Ball(a)$, through their indicator function:

$$\delta_{Box(a)}(x) \equiv \begin{cases} 0 & \|x\|_\infty \leqslant a \\ \infty & \|x\|_\infty > a \end{cases} \tag{9}$$

and

$$\delta_{Ball(a)}(x) \equiv \begin{cases} 0 & \|x\|_2 \leqslant a \\ \infty & \|x\|_2 > a \end{cases}. \tag{10}$$

Recall that the $\|\cdot\|_\infty$ norm selects the largest component of the argument; thus, $Box(a)$ comprises vectors with no component larger than a (in 2D $Box(a)$ is a square centered on the origin with width $2a$). We also employ $\mathbf{0}_X$ and $\mathbf{1}_X$ to mean a vector from the space X with all components set to 0 and 1, respectively.

3.1. Image reconstruction by least-squares

Perhaps the simplest optimization method for performing image reconstruction is to minimize the quadratic data error function. We present this familiar case in order to gain some experience with the mechanics of deriving CP algorithm instances, and because the quadratic data error term will play a role in other optimization problems below. The primal problem of interest is

$$\min_u \tfrac{1}{2} \|Au - g\|_2^2. \tag{11}$$

To derive the CP algorithm instance, we make the following mechanical associations with the primal problem (1):

$$F(y) = \tfrac{1}{2} \|y - g\|_2^2, \tag{12}$$

$$G(x) = 0, \tag{13}$$

$$x = u, \quad y = Au \tag{14}$$

$$K = A. \tag{15}$$

Applying equation (3), we obtain the convex conjugates of F and G:

$$F^*(p) = \tfrac{1}{2} \|p\|_2^2 + \langle p, g \rangle_D, \tag{16}$$

$$G^*(q) = \delta_{\mathbf{0}_I}(q), \tag{17}$$

where $p \in D$ and $q \in I$. While obtaining F^* in this case involves elementary calculus for extremization of equation (3), finding G^* needs some comment for those unfamiliar with convex analysis. Using the definition of the Legendre transform for $G(x) = 0$, we have

$$G^*(q) = \max_x \langle q, x \rangle_I. \tag{18}$$

There are two possibilities: (1) $q = \mathbf{0}_I$, in which case the maximum value of $\langle q, x \rangle_I$ is 0, and (2) $q \neq \mathbf{0}_I$, in which case this inner product can increase without bound, resulting in a maximum value of ∞. Putting these two cases together yields the indicator function in

equation (17). With F, G and their conjugates, the optimization problem dual to equation (11) can be written from equation (2):

$$\max_p \left\{ -\tfrac{1}{2}\|p\|_2^2 - \langle p, g \rangle_D - \delta_{\mathbf{0}_I}(-A^T p) \right\}. \tag{19}$$

For deriving the CP algorithm instance, it is not strictly necessary to have this dual problem, but it is useful for evaluating convergence.

The CP algorithm solves equations (11) and (19) simultaneously. In principle, the values of the primal and dual objective functions provide a test of convergence. During the iteration, the objective of the primal problem will be greater than the objective of the dual problem, and when the solutions of the respective problems are reached, these objectives will be equal. Comparing the duality gap, i.e. the difference between the primal objective and the dual objective, with 0 thus provides a test of convergence. The presence of the indicator function in the dual problem, however, complicates this test. Due to the negative sign in front of the indicator, when the argument is not the zero vector, this term and therefore the whole dual objective is assigned to a value of $-\infty$. The dual objective achieves a finite, testable value only when the indicator function attains the value of 0, when $A^T p = \mathbf{0}_I$. Effectively, the indicator function becomes a way to write down a constraint in the form of a convex function, in this case an equality constraint. The dual optimization problem can thus alternately be written as a conventional constrained maximization

$$\max_p \left\{ -\tfrac{1}{2}\|p\|_2^2 - \langle p, g \rangle_D \right\} \text{ such that } A^T p = \mathbf{0}_I. \tag{20}$$

The convergence check is a bit problematic, because the equality constraint will not likely be strictly satisfied in numerical computation. Instead, we introduce a conditional primal–dual gap (the difference between the primal and dual objectives ignoring the indicator function) given the estimates u' and p':

$$cPD(u', p') = \tfrac{1}{2}\|Au' - g\|_2^2 + \tfrac{1}{2}\|p'\|_2^2 + \langle p', g \rangle_D, \tag{21}$$

and separately monitor $A^T p'$ to see if it is tending to $\mathbf{0}_I$. Note that the conditional primal–dual gap need not be positive, but it should tend to zero.

To finally attain the CP algorithm instance for image reconstruction by least-squares, we derive lines 5 and 6 in algorithm 1. The proximal mapping $prox_\sigma[F^*](y)$, $y \in D$, for this problem results from a quadratic minimization

$$\begin{aligned} prox_\sigma[F^*](y) &= \arg\min_{y'} \left\{ \frac{1}{2}\|y'\|_2^2 + \langle y', g \rangle_D + \frac{\|y - y'\|_2^2}{2\sigma} \right\} \\ &= \frac{y - \sigma g}{1 + \sigma}, \end{aligned} \tag{22}$$

and as $G(x) = 0$, $x \in I$, the corresponding proximal mapping is

$$prox_\tau[G](x) = x. \tag{23}$$

Substituting the arguments from the generic algorithm leads to the update steps in listing 2. The constant $L = \|A\|_2$ is the largest singular value of A (see appendix A for details on the power method). Crucial to the implementation of the CP algorithm instance is that A^T be the exact transpose of A, which is a non-trivial matter for tomographic applications, because the projection matrix A is usually computed on-the-fly (Siddon 1985, De Man and Basu 2004, Xu and Mueller 2007). The convergence of the CP algorithm is only guaranteed when A^T is the exact transpose of A, although it may be possible to extend the CP algorithm to mismatched projector/back-projector pairs by employing the analysis in Zeng and Gullberg (2000).

Algorithm 2. Pseudocode for N steps of the least-squares Chambolle–Pock algorithm instance.

1: $L \leftarrow \|A\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize u_0 and p_0 to zero values
3: $\bar{u}_0 \leftarrow u_0$
4: **repeat**
5:     $p_{n+1} \leftarrow (p_n + \sigma(A\bar{u}_n - g))/(1 + \sigma)$
6:     $u_{n+1} \leftarrow u_n - \tau A^T p_{n+1}$
7:     $\bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
8:     $n \leftarrow n + 1$
9: **until** $n \geqslant N$

This derivation of the CP least-squares algorithm instance illustrates the method on a familiar optimization problem, and it provides a point of comparison with standard algorithms; this quadratic minimization problem can be solved straightforwardly with the basic, linear CG algorithm. Another important point for this particular algorithm instance, where limited projection data can lead to an underdetermined system, is that the CP algorithm will yield a minimizer of the objective $\|Au - g\|_2^2$ which depends on the initial image u_0. In this case, it is recommended to take advantage of the prototyping capability of the CP framework to augment the optimization problem so that it selects a unique image independent of initialization. For example, one often seeks an image closest to either $\mathbf{0}_I$ or a prior image, which can be formulated by adding a quadratic term $\|u\|_2^2$ or $\|u - u_{\text{prior}}\|_2^2$ with a small combination coefficient.

3.1.1. Adding in non-negativity constraints. One of the flexibilities of the CP method becomes apparent in adding bound constraints. While CG is also a flexible tool for dealing with large and small quadratic optimization, modification to include constraints, such as non-negativity, considerably complicates the CG algorithm. For CP, adding in bound constraints is simply a matter of introducing the appropriate indicator function into the primal problem:

$$\min_u \left\{ \tfrac{1}{2}\|Au - g\|_2^2 + \delta_P(u) \right\}, \tag{24}$$

where the set P is all u with non-negative components. Again, we make the mechanical associations with the primal problem (1):

$$F(y) = \tfrac{1}{2}\|y - g\|_2^2, \tag{25}$$

$$G(x) = \delta_P(x), \tag{26}$$

$$x = u, \quad y = Kx, \tag{27}$$

$$K = A. \tag{28}$$

The difference from the unconstrained problem is the function $G(x)$. It turns out that the convex conjugate of $\delta_P(x)$ is

$$\delta_P^*(x) = \delta_P(-x); \tag{29}$$

see appendix B for insight on the convex conjugate of indicator functions. Straight substitution of G^* and F^* into equation (2) yields the dual problem

$$\max_p \left\{ -\tfrac{1}{2}\|p\|_2^2 - \langle p, g \rangle_D - \delta_P(A^T p) \right\}. \tag{30}$$

As a result the conditional primal–dual gap is the same as before. The difference now is that the constraint checks are that $A^T p$ and u should be non-negative.

To derive the algorithm instance, we need the proximal mapping $prox_\tau[G]$, which by definition is

$$prox_\tau[\delta_P](x) = \arg\min_{x'} \left\{ \delta_P(x') + \frac{\|x - x'\|_2^2}{2\tau} \right\}. \tag{31}$$

The indicator in the objective prevents consideration of negative components of x'. The ℓ_2 term can be regarded as a sum over the square difference between components of x and x'; thus, the objective is separable and can be minimized by constructing x' such that $x_i' = x_i$ when $x_i > 0$ and $x_i' = 0$ when $x_i \leqslant 0$. Thus, this proximal mapping becomes a non-negativity thresholding on each component of x:

$$[prox_\tau[\delta_P](x)]_i \equiv [pos(x)]_i \equiv \begin{cases} x_i & x_i > 0 \\ 0 & x_i \leqslant 0 \end{cases}. \tag{32}$$

Substituting into the generic pseudocode yields listing 3. Again, we have $L = \|A\|_2$. The indicator function δ_P leads to the intuitive modification that non-negativity thresholding is introduced in line 6 of listing 3. In this case, the non-negativity constraint in u will be automatically satisfied by all iterates u_n. Upper bound constraints are equally simple to include.

Algorithm 3. Pseudocode for N steps of the least squares with the non-negativity constraint, CP algorithm instance.

1: $L \leftarrow \|A\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize u_0 and p_0 to zero values
3: $\bar{u}_0 \leftarrow u_0$
4: **repeat**
5: $\quad p_{n+1} \leftarrow (p_n + \sigma(A\bar{u}_n - g))/(1 + \sigma)$
6: $\quad u_{n+1} \leftarrow pos(u_n - \tau A^T p_{n+1})$
7: $\quad \bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
8: $\quad n \leftarrow n + 1$
9: **until** $n \geqslant N$

3.2. Optimization problems based on the TV semi-norm. Optimization problems with the TV semi-norm have received much attention for CT image reconstruction lately because of their potential to provide high quality images from sparse view sampling (Sidky *et al* 2010, 2011, Bian *et al* 2010, Choi *et al* 2010, Ritschl *et al* 2011, Han *et al* 2011, Xia *et al* 2011). The TV semi-norm has been known to be useful for performing edge-preserving regularization, and recent developments in compressive sensing (CS) have sparked even greater interest in the use of this semi-norm. Algorithmwise the TV semi-norm is difficult to handle. Although it is convex, it is not linear, quadratic or even everywhere-differentiable, and the lack of differentiability precludes the use of standard gradient-based optimization algorithms. In this sub-section, we go through, in detail, the derivation of a CP algorithm instance for a TV-regularized least-squares data error norm. We then consider the Kullback–Leibler (KL) data divergence, which is implicitly employed by many iterative algorithms based on maximum likelihood expectation maximization (MLEM). We also consider a data error norm based on ℓ_1 which can have some advantage in reducing the impact of image discretization error, which generally leads to a highly non-uniform error in the data domain. Finally, we derive a CP

algorithm instance for constrained TV-minimization, which is mathematically equivalent to the least-squares-plus-TV problem (Elad 2010), but whose data-error constraint parameter has more physical meaning than the parameter used in the corresponding unconstrained minimization. While the previous CP instances solve optimization problems, which can be solved efficiently by well-known algorithms, the following CP instances are new for the application of CT image reconstruction.

The optimization problem of interest is

$$\min_u \left\{ \tfrac{1}{2}\|Au - g\|_2^2 + \lambda \|(|\nabla u|)\|_1 \right\}, \tag{33}$$

where the last term, the ℓ_1-norm of the gradient-magnitude image, is the isotropic TV semi-norm. The spatial-vector image ∇u represents a discrete approximation to the image gradient which is in the vector space V, i.e. the space of spatial-vector-valued image arrays. The expression $|\nabla u|$ is the gradient-magnitude image, an image array whose pixel values are the gradient magnitude at the pixel location. Thus, $\nabla u \in V$ and $|\nabla u| \in I$. Because ∇ is defined in terms of finite differencing, it is a linear transform from an image array to a vector-valued image array, the precise form of which is covered in appendix D. This problem was not explicitly covered in Chambolle and Pock (2011), and we fill in the details here. For this case, matching the primal problem to equation (1) is not as obvious as the previous examples. We recognize in equation (33) that both terms involve a linear transform; thus, the whole objective function can be written in the form $F(Kx)$ with the following assignments:

$$F(y, z) = F_1(y) + F_2(z), \quad F_1(y) = \frac{1}{2}\|y - g\|_2^2, \quad F_2(z) = \lambda \|(|z|)\|_1, \tag{34}$$

$$G(x) = 0, \tag{35}$$

$$x = u, \quad y = Au, \quad z = \nabla u, \tag{36}$$

$$K = \begin{pmatrix} A \\ \nabla \end{pmatrix}, \tag{37}$$

where $u \in I$, $y \in D$ and $z \in V$. Note that $F(y, z)$ is convex because it is the sum of two convex functions. Also the linear transform K takes an image vector x and gives a data vector y and an image gradient vector z. The transpose of K, $K^T = (A^T, -\text{div})$, will produce an image vector from a data vector y and an image gradient vector z:

$$x \leftarrow A^T y - \text{div}\, z, \tag{38}$$

where we use the same convention as in Chambolle and Pock (2011) that $-\text{div} \equiv \nabla^T$; see appendix D.

In order to get the convex conjugate of F we need F_2^*. For readers unfamiliar with the Legendre transform of indicator functions, appendix B illustrates the transform of some common cases. By definition,

$$F_2^*(q) = \max_z \left\{ \langle q, z \rangle_V - \lambda \|(|z|)\|_1 \right\}, \tag{39}$$

where $q \in V$, like z, is a vector-valued image array. There are two cases to be considered: (1) the magnitude image $|q|$ at all pixels is less than or equal to λ, i.e. $|q| \in Box(\lambda)$ and (2) the magnitude image $|q|$ has at least one pixel greater than λ, i.e. $|q| \notin Box(\lambda)$. It turns out that for the former case, the maximization in equation (39) yields 0, while the latter case yields ∞. Putting these two cases together, we have

$$F_2^*(q) = \delta_{\text{Box}(\lambda)}(|q|). \tag{40}$$

The conjugates of F and G are

$$F^*(p, q) = \tfrac{1}{2}\|p\|_2^2 + \langle p, g\rangle_D + \delta_{\text{Box}(\lambda)}(|q|), \tag{41}$$

$$G^*(r) = \delta_{\mathbf{0}_I}(r), \tag{42}$$

where $p \in D$, $q \in V$ and $r \in I$.

The problem dual to equation (33) becomes

$$\max_{p,q} \ \left\{ -\tfrac{1}{2}\|p\|_2^2 - \langle p, g\rangle_D - \delta_{\text{Box}(\lambda)}(|q|) - \delta_{\mathbf{0}_I}(\text{div}\, q - A^T p) \right\}. \tag{43}$$

The resulting conditional primal–dual gap is

$$cPD(u', p', q') = \tfrac{1}{2}\|Au' - g\|_2^2 + \lambda\|(|\nabla u'|)\|_1 + \tfrac{1}{2}\|p'\|_2^2 + \langle p', g\rangle_P \tag{44}$$

with additional constraints $|q'| \in \text{Box}(\lambda)$ and $A^T p' - \text{div}\, q' = \mathbf{0}_I$. The final piece needed for putting together the CP algorithm instance for equation (33) is the proximal mapping

$$prox_\sigma[F^*](y, z) = \left(\frac{y - \sigma g}{1 + \sigma}, \frac{\lambda z}{\max(\lambda \mathbf{1}_I, |z|)} \right). \tag{45}$$

The proximal mapping of the data term was covered previously, and that of the TV term is explained in appendix C. With the necessary pieces in place, the CP algorithm instance for the ℓ_2^2-TV objective can be written in listing 4. Line 6 and the corresponding expression in equation (45) require some explanation, because the division operation is non-standard as the numerator is in V and the denominator is in I. The effect of this line is to threshold the magnitude of the spatial vectors at each pixel in $q_n + \sigma \nabla \bar{u}_n$ to the value λ: spatial vectors larger than λ have their magnitude rescaled to λ. The resulting thresholded spatial-vector image is then assigned to q_{n+1}. Recall that $\mathbf{1}_I$ in line 6 is an image with all pixels set to 1. The operator $|\cdot|$ in this line converts a vector-valued image in V to a magnitude image in I, and the $\max(\lambda \mathbf{1}_I, \cdot)$ operation thresholds the lower bound of the magnitude image to λ pixelwise. Operationally, the division is performed by dividing the spatial vector at each pixel of the numerator by the scalar in the corresponding pixel of the denominator. Another potential source of confusion is computing the magnitude $\|(A, \nabla)\|_2$. The power method for doing this is covered explicitly in appendix A. If it is desired to enforce the positivity constraint, the indicator $\delta_P(u)$ can be added to the primal objective, and the effect of this indicator is the same as for listing 3; namely, the right-hand side of line 7 goes inside the $pos(\cdot)$ operator.

Algorithm 4. Pseudocode for N steps of the ℓ_2^2-TV CP algorithm instance.

1:   $L \leftarrow \|(A, \nabla)\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2:   initialize u_0, p_0, and q_0 to zero values
3:   $\bar{u}_0 \leftarrow u_0$
4:   **repeat**
5:      $p_{n+1} \leftarrow (p_n + \sigma(A\bar{u}_n - g))/(1 + \sigma)$
6:      $q_{n+1} \leftarrow \lambda(q_n + \sigma \nabla \bar{u}_n)/\max(\lambda \mathbf{1}_I, |q_n + \sigma \nabla \bar{u}_n|)$
7:      $u_{n+1} \leftarrow u_n - \tau A^T p_{n+1} + \tau\, div\, q_{n+1}$
8:      $\bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
9:      $n \leftarrow n + 1$
10: **until** $n \geqslant N$

3.2.1. Alternate data divergences. For a number of reasons motivated by the physical model of imaging systems, it may be of use to formulate optimization problems for CT image reconstruction with alternate data-error terms. A natural extension of the quadratic data divergence is to include a diagonal weighting matrix. The corresponding CP algorithm instance can be easily derived following the steps mentioned above. As pointed out above, the CP method is not limited to quadratic objective functions and other important convex functions can be used. We derive, here, two additional CP algorithm instances. For alternate data divergences we consider the oft-used KL divergence, and one not so commonly used ℓ_1 data-error norm. For the following, we need only analyze the function F_1, as everything else remains the same as for the ℓ_2^2-TV objective in equation (33).

TV plus KL data divergence One data divergence of particular interest for tomographic image reconstruction is KL. Objectives based on KL are what is being optimized in the various forms of MLEM, and it is used often when data noise is a significant physical factor and the data are modeled as being drawn from a multivariate Poisson probability distribution (Barrett and Myers 2004). For the situation where the view sampling is also sparse, it might be of interest to combine a KL data error term with the TV semi-norm in the following primal optimization:

$$\min_u \left\{ \sum_i [Au - g + g \ln g - g \ln(pos(Au))]_i + \delta_P(Au) + \lambda \left\| (|\nabla u|) \right\|_1 \right\}, \tag{46}$$

where $\sum_i [\cdot]_i$ performs summation over all components of the vector argument. This example proceeds as above except that the F_1 function is different:

$$F_1(y) = \sum_i [y - g + g \ln g - g \ln(pos(y))]_i + \delta_P(y), \tag{47}$$

where $y \in D$ and the function ln operates on the components of its argument. The use of the KL data divergence makes sense only with positive linear systems A and non-negative pixel values u and data g. However, by defining the function over the whole space and using an indicator function to restrict the domain (Rockafellar *et al* 1970), a wide variety of optimization problems can be treated in a uniform manner. Accordingly, δ_P is introduced into the F_1 objective and the *pos* operator is used just so that this objective is defined in the real numbers. The derivation of F_1^*, though mechanical, is a little bit too long to be included here. We simply state the resulting conjugate function:

$$F_1^*(p) = \sum_i [-g \ln pos(\mathbf{1}_D - p)]_i + \delta_P(\mathbf{1}_D - p). \tag{48}$$

The resulting dual problem to equation (46) is thus

$$\max_{p,q} \left\{ \sum_i [g \ln pos(\mathbf{1}_D - p)]_i - \delta_P(\mathbf{1}_D - p) - \delta_{\text{Box}(\lambda)}(|q|) - \delta_{\mathbf{0}_I}(div\, q - A^T p) \right\}. \tag{49}$$

To form the algorithm instance, we need the proximal mapping $prox_\sigma[F_1^*](y)$

$$prox_\sigma[F_1^*](y) = \tfrac{1}{2}(\mathbf{1}_D + y - \sqrt{(y - \mathbf{1}_D)^2 + 4\sigma g}). \tag{50}$$

An interesting point in the derivation, shown partially in appendix C, of $prox_\sigma[F_1^*](y)$ is that the quadratic equation is needed, and the indicator function in $F_1^*(p)$ is used to select the correct (in this case negative) root of the discriminant in the quadratic formula. With the new function F_1, its conjugate and the conjugate's proximal mapping, we can write down the CP algorithm instance. Listing 5 gives the CP algorithm instance minimizing a KL plus TV semi-norm objective.The difference between this algorithm instance and the previous ℓ_2^2-TV

case comes only at the update at line 5. This algorithm instance has the interesting property that the intermediate image estimates u_n can have negative values even though the converged solution will be non-negative. If it is desirable to have the intermediate image estimates be non-negative, the non-negativity constraint can be easily introduced by adding the indicator $\delta_P(u)$ to the primal objective, resulting in the addition of the $pos(\cdot)$ operator at line 7 as was shown in listing 3.

Algorithm 5. Pseudocode for N steps of the KL-TV CP algorithm instance.

1: $L \leftarrow \|(A, \nabla)\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize u_0, p_0, and q_0 to zero values
3: $\bar{u}_0 \leftarrow u_0$
4: **repeat**
5: $\quad p_{n+1} \leftarrow \frac{1}{2}\left(\mathbf{1}_D + p_n + \sigma A\bar{u}_n - \sqrt{(p_n + \sigma A\bar{u}_n - \mathbf{1}_D)^2 + 4\sigma g}\right)$
6: $\quad q_{n+1} \leftarrow \lambda(q_n + \sigma\nabla\bar{u}_n)/\max(\lambda\mathbf{1}_I, |q_n + \sigma\nabla\bar{u}_n|)$
7: $\quad u_{n+1} \leftarrow u_n - \tau A^T p_{n+1} + \tau\, div\, q_{n+1}$
8: $\quad \bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
9: $\quad n \leftarrow n + 1$
10: **until** $n \geqslant N$

TV plus ℓ_1 data-error norm   The combination of TV semi-norm regularization and ℓ_1 data-error norm has been proposed for image denoising and it has some interesting properties for that purpose (Chan and Esedoglu 2005). This objective is also presented in Chambolle and Pock (2011). For tomography, this combination may be of interest because the ℓ_1 data-error term is an example of a robust fit to the data. The idea of robust approximation is to weakly penalize data that are outliers (Boyd and Vandenberghe 2004). Fitting with the commonly used quadratic error function clearly puts heavy weight on outlying measurements which in some situations can lead to streak artifacts in the images. In particular, for tomographic image reconstruction with a pixel basis, discretization error and metal objects can lead to highly non-uniform error in the data model. The use of the ℓ_1 data-error term may allow for large errors for measurements along the tangent rays to internal structures, where discretization can have a large effect. The ℓ_1 data-error term also puts greater weight on driving small data errors towards zero. The primal problem of interest is

$$\min_u \{\|Au - g\|_1 + \lambda\|(|\nabla u|)\|_1\}. \tag{51}$$

For this objective, the function F_1 is

$$F_1(y) = \|y - g\|_1. \tag{52}$$

Computing the convex conjugate F_1^* yields

$$F_1^*(p) = \delta_{\text{Box}(1)}(p) + \langle p, g\rangle_D, \tag{53}$$

and the resulting dual problem is

$$\max_{p,q}\{-\delta_{\text{Box}(1)}(p) - \langle p, g\rangle_D - \delta_{\text{Box}(\lambda)}(|q|) - \delta_{\mathbf{0}_I}(\text{div}\, q - A^T p)\}. \tag{54}$$

The proximal mapping necessary for completing the algorithm instance is

$$prox_\sigma[F_1^*](y) = \frac{y - g\sigma}{\max(\mathbf{1}_D, |y - g\sigma|)}, \tag{55}$$

where $\mathbf{1}_D$ is a data array with each component set to 1 and the max operation is performed componentwise. The corresponding pseudo-code for minimizing equation (51) is given in listing 6, where the only difference between this code and the previous two occurs at line 5. The ability to deal with non-smooth objectives uncomplicates this particular problem substantially. If smoothness were required, there would have to be smoothing parameters on both the ℓ_1 and TV terms, adding two more parameters than necessary to a study of the image properties as a function of the optimization-problem parameters.

Algorithm 6. Pseudocode for N steps of the ℓ_1-TV CP algorithm instance.

1: $L \leftarrow \|(A, \nabla)\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize u_0, p_0, and q_0 to zero values
3: $\bar{u}_0 \leftarrow u_0$
4: **repeat**
5:      $p_{n+1} \leftarrow (p_n + \sigma (A\bar{u}_n - g))/ \max(\mathbf{1}_D, |p_n + \sigma (A\bar{u}_n - g)|)$
6:      $q_{n+1} \leftarrow \lambda(q_n + \sigma \nabla \bar{u}_n)/ \max(\lambda \mathbf{1}_I, |q_n + \sigma \nabla \bar{u}_n|)$
7:      $u_{n+1} \leftarrow u_n - \tau A^T p_{n+1} + \tau \, div \, q_{n+1}$
8:      $\bar{u}_{n+1} \leftarrow u_{n+1} + \theta (u_{n+1} - u_n)$
9:      $n \leftarrow n + 1$
10: **until** $n \geqslant N$

3.2.2. Constrained, TV-minimization. The previous three optimization problems combine a data fidelity term with a TV penalty, and the balance of the two terms is controlled by the parameter λ. An inconvenience of such optimization problems is that it is difficult to physically interpret λ. Focusing on combining an ℓ_2 data-error norm with TV, reformulating equation (33) as a constrained, TV-minimization leads to the following primal problem:

$$\min_u \{\|(|\nabla u|)\|_1 + \delta_{\mathrm{Ball}(\epsilon)}(Au - g)\}, \tag{56}$$

where $\delta_{\mathrm{Ball}(\epsilon)}(Au - g)$ is zero for $\|Au - g\|_2 \leqslant \epsilon$. When $\epsilon > 0$, this problem is equivalent to the unconstrained optimization (33), see e.g. Elad (2010), in the sense that for each positive ϵ there is a corresponding λ yielding the same solution. For this constrained, TV-minimization, the function F_1 is

$$F_1(y) = \delta_{\mathrm{Ball}(\epsilon)}(y - g). \tag{57}$$

The corresponding conjugate is

$$F_1^*(p) = \epsilon \|p\|_2 + \langle p, g \rangle_D, \tag{58}$$

leading to the dual problem

$$\max_{p,q} \{-\epsilon \|p\|_2 - \langle p, g \rangle_D - \delta_{\mathrm{Box}(1)}(|q|) - \delta_{\mathbf{0}_I}(\mathrm{div}\, q - A^T p)\}. \tag{59}$$

Again for the algorithm instance we need the proximal mapping $prox_\sigma[F_1^*]$:

$$prox_\sigma[F_1^*](y) = \max(\|y - \sigma g\|_2 - \sigma \epsilon, 0)\, (y - \sigma g). \tag{60}$$

The main points in deriving this proximal mapping are discussed in appendix C, and it is an example where geometric/symmetry arguments play a large role. Listing 7 shows the algorithm instance solving equation (56), where once again only line 5 is modified.

This algorithm instance essentially achieves the same goal as listing 4; the only difference is that the parameter ϵ has an actual physical interpretation, being the data-error bound.

Algorithm 7. Pseudocode for N steps of the ℓ_2-constrained, TV-minimization CP algorithm instance.

1: $L \leftarrow \|(A, \nabla)\|_2$; $\tau \leftarrow 1/L$; $\sigma \leftarrow 1/L$; $\theta \leftarrow 1$; $n \leftarrow 0$
2: initialize u_0, p_0, and q_0 to zero values
3: $\bar{u}_0 \leftarrow u_0$
4: **repeat**
5: $\quad p_{n+1} \leftarrow \max(\|p_n + \sigma(A\bar{u}_n - g)\|_2 - \sigma\epsilon, 0)(p_n + \sigma(A\bar{u}_n - g))$
6: $\quad q_{n+1} \leftarrow (q_n + \sigma\nabla\bar{u}_n)/\max(\mathbf{1}_I, |q_n + \sigma\nabla\bar{u}_n|)$
7: $\quad u_{n+1} \leftarrow u_n - \tau A^T p_{n+1} + \tau \operatorname{div} q_{n+1}$
8: $\quad \bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
9: $\quad n \leftarrow n + 1$
10: **until** $n \geqslant N$

---

4. Demonstration of CP algorithm instances for tomographic image reconstruction

In the previous section, we have derived CP algorithm instances covering many optimization problems of interest to CT image reconstruction. Not only are there the seven optimization problems, but within each case the system model/matrix A, the data g and optimization problem parameters can vary. For each of these, practically infinite number of optimization problems, the corresponding CP algorithm instances are guaranteed to converge (Chambolle and Pock 2011). The purpose of this results section is not to advocate one optimization problem over another, but rather to demonstrate the utility of the CP algorithm for optimization problem prototyping. For this purpose, we present example image reconstructions that could be performed in a study for investigating the impact of matching the data divergence with the data noise model for image reconstruction in breast CT.

4.1. Experiments for sparse-view image reconstruction from simulated CT data

We briefly describe the significance of the experiments, but we point out that the main goal here is to demonstrate the CP algorithm instances. Much of the recent interest in employing the TV semi-norm in optimization problems for CT image reconstruction has been generated by CS. CS seeks to relate sampling conditions on a sensing device with sparsity in the object being scanned. So far, mathematical results have been limited to various types of random sampling (Candès and Wakin 2008). System matrices such as those representing CT projection fall outside of the scope of mathematical results for CS (Sidky *et al* 2010). As a result, the only current option for investigating CS in CT is through numerical experiments with computer phantoms.

A next logical step for bridging theoretical results for CS to actual application is to consider physical factors in the data model. One such factor is a noise model, which can be quite important for low-dose CT applications such as breast CT. While much work has been performed on iterative image reconstruction with various noise models under conditions of full sampling, little is known about the impact of noise on sparse-view image reconstruction. In the following limited study, we set up a breast CT simulation to investigate the impact
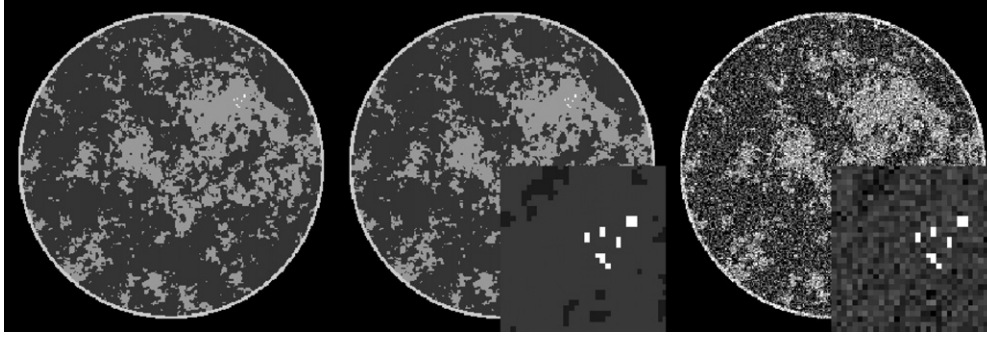
Figure 1. Breast phantom for CT and FBP reconstructed image for a 512-view dataset with Poisson distributed noise. Left: the phantom in the gray scale window [0.95,1.15]; middle: the same phantom with a blow-up on the micro-calcification ROI displayed in the gray scale window [0.9,1.8]; right: the FBP image reconstructed from the noisy data. The middle panel is the reference for all image reconstruction algorithm results. The FBP image is shown only to provide a sense of the noise level.

of correct modeling of data noise with the purpose of demonstrating that the CP algorithm instances can be applied to the CT system.

4.2. Sparse-view reconstruction with a Poisson noise model

For the following study, we employ a digital 256×256 breast phantom, described in Jørgensen *et al* (2011b), Reiser and Nishikawa (2010), and used in our previous study on investigating sufficient sampling conditions for TV-based CT image reconstruction (Jørgensen *et al* 2011a). The phantom models four tissue types: the background fat tissue is assigned a value of 1.0, the modeled fibro-glandular tissue takes a value of 1.1, the outer skin layer is set to 1.15, and the micro-calcifications are assigned values in the range [1.8,2.3].

For the present case, we focus on circular, fan-beam scanning with 60 projections equally distributed over a full $360°$ angular range. The simulated radius of the x-ray source trajectory is 40 cm with a source–detector distance of 80 cm. The detector sampling consists of 512 bins of size 200 μm. The system matrix for the x-ray projection is computed by the line-intersection method where the matrix elements of *A* are determined by the length of traversal in each image pixel of each source/detector bin ray. For this phantom under ideal conditions, we have found that an accurate recovery is possible with constrained, TV-minimization with as few as 50 projections. In this study, we add Poisson noise to the data model at a level consistent with what might be expect in a typical breast CT scan. The Poisson noise model is chosen in order to investigate the impact of matching the data-error term to the noise model. For reference, the phantom is shown in figure 1. To have a sense of the noise level, a standard fan-beam filtered back-projection reconstruction is shown alongside the phantom for simulated Poisson noise.

For this noise model, the maximum likelihood method prescribes minimizing the KL data divergence between the available and estimated data. To gauge the importance of selecting a maximum likelihood image, we compare the results from two optimization problems: a KL data divergence plus a TV-penalty, equation (46) above; and a least-squares data error norm plus a TV-penalty, equation (33) above. With the CP framework, these two optimization problems can be easily prototyped: the solutions to both problems can be obtained without worrying about smoothing the TV semi-norm, setting algorithm parameters or proving convergence.
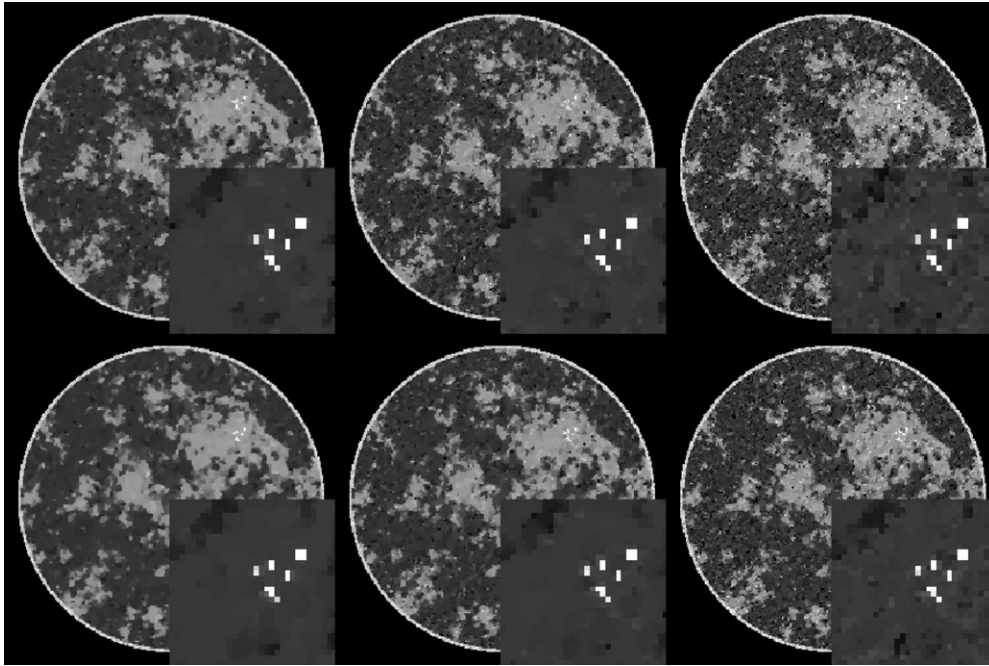
Figure 2. Images reconstructed from 60-view projection data with a Poisson-distributed noise model. The top row of images result from minimizing the ℓ_2^2-TV objective in equation (33) for $\lambda = 1 \times 10^{-4}, 5 \times 10^{-5}$ and 2×10^{-5}, going from left to right. The bottom row of images result from minimizing the KL-TV objective in equation (46) for the same values of λ. Note that λ does not necessarily have the same impact on each of these optimization problems. Nevertheless, we see similar trends for the chosen values of λ.

For the phantom and data conditions, described above, the images for different values of the TV-penalty parameter λ are shown in figure 2. An ROI of the micro-calcification cluster is also shown. The overall and ROI images give an impression of two different visual tasks important for breast imaging: discerning the fibro-glandular tissue morphology and detection/classification of micro-calcifications. The images show some difference between the two optimization problems; most notably there is a perceptible reduction in noise in the ROIs from the KL-TV images. A firm conclusion, however, awaits a more complete study with multiple noise realizations.

The most critical feature of the CP algorithm that we wish to promote is the rapid prototyping of a convex optimization problem for CT image reconstruction. The above study is aimed at a combination of using a data divergence based on maximum likelihood estimation with a TV penalty, which takes advantage of sparsity in the gradient magnitude of the underlying object. The CP framework facilitates the use of many other convex optimization problems, particularly those based on some form of sparsity, which often entail some form of the non-smooth ℓ_1-norm. For example, in Sidky *et al* (2010), we have found it useful for sparse-view x-ray phase-contrast imaging to perform image reconstruction with a combination of a least-squares data fidelity term, an ℓ_1-penalty promoting object sparseness, and an image TV constraint to further reduce streak artifacts from angular under-sampling. Under the CP framework, prototyping various combinations of these terms as constrained or unconstrained optimization problems becomes possible and the corresponding derivation of CP algorithm
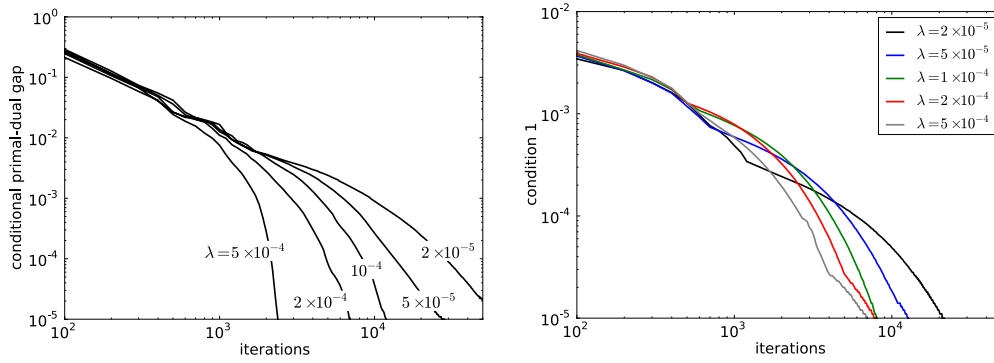
Figure 3. Left: convergence of the conditional primal–dual gap for the CP algorithm instance solving equation (46) for different values of λ. Right: plot indicating agreement with condition 1: $\|div\, q - A^T p\|_\infty$, the magnitude of the largest component of the argument of the last indicator function of equation (49). Collecting all the indicator functions of the primal, equation (46), and dual, equation (49), KL-TV optimization problems, we have four conditions to check in addition to the conditional primal–dual gap: (1) $div\, q - A^T p = \mathbf{0}_I$, (2) $Au \geqslant \mathbf{0}_D$, (3) $p \leqslant \mathbf{1}_D$ and (4) $|q| < \lambda$. The agreement with condition 1 is illustrated in the plot; agreement with condition 2 has a similar dependence; condition 3 is satisfied early on in the iteration; and condition 4 is automatically enforced by the CP algorithm instance for KL-TV. Because the curves are bunched together in the condition 1 plot, they are differentiated in color.

instances follows from the steps described in section 2.2. Alternative, convex data fidelity terms and image constraints motivated by various physical models may also be prototyped.

As a practical matter, though, it is important to have some sense of the convergence of the CP algorithm instances. To this end, we take an in-depth look at individual runs for the KL-TV algorithm instance for CT image reconstruction.

4.3. Iteration dependence of the CP algorithm

Through the methods described above, many useful algorithm instances can be derived for CT image reconstruction. It is obviously important that the resulting algorithm instance reaches the solution of the prescribed optimization problem. To illustrate the convergence of a resulting algorithm instance, we focus on the TV-penalized KL data divergence, equation (46), and plot the conditional primal–dual gap for the different runs with varying λ in figure 3. Included in this figure is a plot indicating the convergence to agreement with the most challenging condition set by the indicator functions in equation (49). For the present results, we terminated the iteration at a conditional primal–dual gap of 10^{-5}, which appears to happen on the scale of thousands of iterations with smaller λ requiring more iterations. Interestingly, a simple pre-conditioned form of the CP algorithm was proposed in Pock and Chambolle (2011), which appears to perform efficiently for small λ. The pre-conditioned CP algorithm instance for this problem is reported in appendix E.

5. Discussion

This paper has presented the application of the CP algorithm to prototyping of optimization problems for CT image reconstruction. The algorithm covers many optimization problems of

interest allowing for non-smooth functions. It also comes with solid convergence criteria to check the image estimates.

The use of the CP algorithm we are promoting here is for prototyping, namely when the image reconstruction algorithm development is at the early stage of determining important factors in formulating the optimization problem. As an example, we illustrated a scenario for sparse-view breast CT considering two different data-error terms. In this stage of development, it is helpful to not have to bother with algorithm parameters, and questions of whether or not the algorithm will converge. After the final optimization problem is determined, the focus shifts from prototyping to efficiency.

Optimization problem prototyping for CT image reconstruction does have its limitations. For example, in the breast CT simulation presented above, a more complete conclusion requires reconstruction from multiple realizations of the data under the Poisson noise model. Additional important dimensions of the study are generation of an ensemble of breast phantoms and considering alternate image representations/projector models. Considering the size of CT image reconstruction systems and huge parameter space of possible optimization problems, it is not yet realistic to completely characterize a particular CT system. But at least we are assured of solving isolated setups and it is conceivable to perform a study along one aspect of the system, i.e. consider multiple realizations of the random data model. Given the current state of affairs for optimization-based image reconstruction, it is crucial that simulations be as realistic as possible. There is great need for realistic phantoms, and data simulation software.

We point out that it is likely at least within the immediate future that optimization-based image reconstruction will have to operate at severely truncated iteration numbers. Current clinical applications of iterative image reconstruction often operate in the range of one to ten iterations, which is likely far too few for claiming that the image estimate is an accurate solution to the designed optimization problem. But at least the ability to prototype an optimization problem can potentially simplify the design phase by separating optimization parameters from algorithm parameters.

Acknowledgments

Appendix A. Computing the norm of K

The matrix norm used for the parameter L in the CP algorithm instances is the largest singular value of K. This singular value can be obtained by the standard power method specified in listing 8. When K represents the discrete x-ray transform, our experience has been that the power method converges to numerical precision in 20 iterations or less. In implementing the CP algorithm instance for TV-penalized minimization, the norm of the combined linear transform $\|(A, \nabla)\|_2$ is needed. For this case, the program is the same as listing 8 where $K^T K x_n$ becomes $A^T A x_n - \text{div} \nabla x_n$; recall that $-\text{div} = \nabla^T$. Furthermore, to obtain s, the explicit computation is $s = \sqrt{\|A x_{n+1}\|_2^2 + \|\nabla x_{n+1}\|_2^2}$.
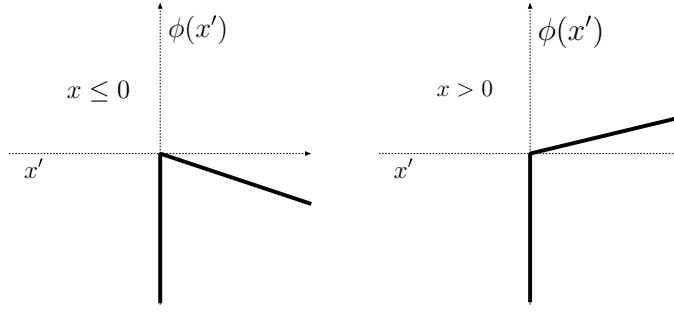
Figure B1. Illustration of the objective function, labeled $\phi(x')$, in the maximization described by equation (B.1). Shown are the two cases discussed in the text.

Algorithm 8. Pseudocode for N steps of the generic power method. The scalar s tends to $\|K\|_2$ as N increases.

1: initialize $x_0 \in I$ to a non-zero image
2: $n \leftarrow 0$
3: **repeat**
4:   $x_{n+1} \leftarrow K^T K x_n$
5:   $x_{n+1} \leftarrow x_{n+1}/\|x_{n+1}\|_2$
6:   $s \leftarrow \|K x_{n+1}\|_2$
7:   $n \leftarrow n + 1$
8: **until** $n \geqslant N$

Appendix B. The convex conjugate of certain indicator functions of interest illustrated in one dimension

This appendix covers the convex conjugate of a couple of indicator functions in one dimension, serving to illustrate how geometry plays a role in the computation and to provide a mental picture on the conjugate of higher-dimensional indicator functions.

Consider first the indicator $\delta_P(x)$, which is zero for $x \geqslant 0$. The conjugate of this indicator is computed from

$$\delta_P^*(x) = \max_{x'} \phi(x') = \max_{x'}\{x'x - \delta_P(x')\}. \tag{B.1}$$

To perform this maximization, we analyze the cases, $x \leqslant 0$ and $x > 0$, separately. As a visual aid, we plot the objective for these two cases in figure B1. From this figure it is clear that when $x \leqslant 0$, the objective's maximum is attained at $x' = 0$ and this maximum value is 0 (note that this is true even for $x = 0$). When $x > 0$, the objective can increase without bound as x' tends to ∞, resulting in a maximum value of ∞. Putting these two cases together yields

$$\delta_P^*(x) = \delta_P(-x).$$

Generalizing this argument to multi-dimensional x yields equation (29).
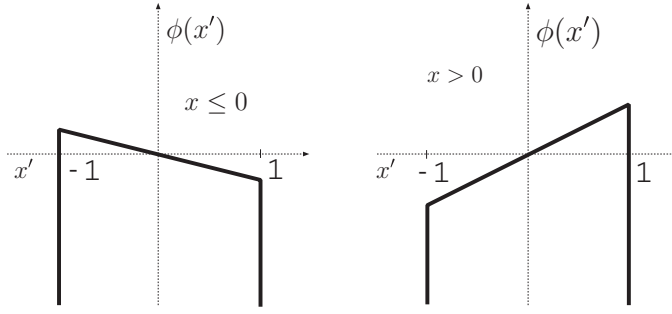
Figure B2. Illustration of the objective function, labeled $\phi(x')$, in the maximization described by equation (B.2). Shown are the two cases discussed in the text.

Next we consider $\delta_{\mathrm{Box}(1)}(x)$, which in one dimension is the same as $\delta_{\mathrm{Ball}(1)}(x)$. This function is zero only for $-1 \leqslant x \leqslant 1$. Its conjugate is computed from

$$\delta_{\mathrm{Box}(1)}^*(x) = \max_{x'} \ \phi(x') = \max_{x'}\{x'x - \delta_{\mathrm{Box}(1)}(x')\}. \tag{B.2}$$

Again we have two cases, $x \leqslant 0$ and $x > 0$, illustrated in figure B2. In the former case, the maximum value of the objective is attained at $x' = -1$, and this maximum value is $-x$. In the latter case, the maximum value is x, and it is attained at $x' = 1$. Hence, we have

$$\delta_{\mathrm{Box}(1)}^*(x) = |x|.$$

For multi-dimensional x, $\delta_{\mathrm{Box}(1)}(x) \neq \delta_{\mathrm{Ball}(1)}(x)$, and this is also reflected in the conjugates

$$\delta_{\mathrm{Box}(1)}^*(x) = \|x\|_1,$$
$$\delta_{\mathrm{Ball}(1)}^*(x) = \|x\|_2.$$

It is also interesting to verify that $\delta_{\mathrm{Box}(1)}^{**}(x)$ is indeed $\delta_{\mathrm{Box}(1)}(x)$ by showing, again in one dimension, that $|x|^* = \delta_{\mathrm{Box}(1)}(x)$. Illustrating this example helps in understanding the convex conjugate of multi-dimensional ℓ_1-based semi-norms. The relevant conjugate is computed from

$$|x|^* = \max_{x'} \phi(x') = \max_{x'}\{x'x - |x'|\}. \tag{B.3}$$

Here, we need to analyze three cases: $x < -1$, $-1 \leqslant x \leqslant 1$ and $x > 1$. The corresponding sketch is in figure B3. The $-|x|$ term in the objective makes an upside-down wedge, and the $x'x$ term serves to tip this wedge. In the second case, the wedge is tipped, but still opens up downward so that the objective is maximized at $x' = 0$, attaining there the value of 0. In the first and third cases, the wedge is tipped so much that part of it points upward and the objective can increase without bound, attaining the value of ∞. Putting these cases together does indeed yield

$$|x|^* = \delta_{\mathrm{Box}(1)}(x).$$

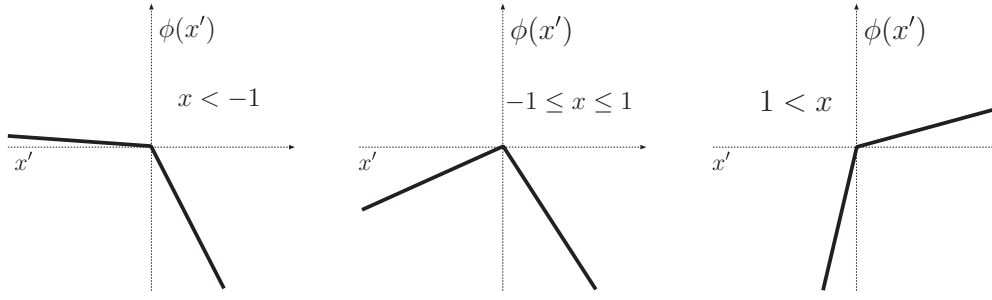A similar reasoning is used to obtain equation (40) from equation (39).

Figure B3. Illustration of the objective function, labeled $\phi(x')$, in the maximization described by equation (B.3). Shown are the three cases discussed in the text.

Appendix C. Computation of important proximal mappings

This appendix fills in important steps in computing some of the proximal mappings in the text, where it is necessary to use geometrical reasoning in addition to setting the gradient of the objective to zero.

The conjugate of the TV semi-norm in equation (40) leads to the following proximal mapping computation:

$$prox_\sigma[F^*](z) = \arg\min_{z'} \left\{ \delta_{\text{Box}(\lambda)}(|z'|) + \frac{\|z - z'\|_2^2}{2\sigma} \right\},$$

where $z, z' \in V$, and the absolute value, $|\cdot|$, of a spatial-vector image V yields an image, in I, of the spatial-vector magnitude. The quadratic term is minimized when $z = z'$, but the indicator function excludes this minimizer when $|z| \notin Box(\lambda)$. To solve this problem, we write the quadratic as a sum over pixels:

$$\frac{\|z - z'\|_2^2}{2\sigma} = \frac{\sum_i |z_i - z_i'|^2}{2\sigma},$$

where i indexes the image pixels and each z_i and z_i' is a spatial-vector. The indicator function places an upper bound on the magnitude of each spatial vector $|z_i'| \leqslant \lambda$. The proximal mapping is built pixel-by-pixel considering two cases: if $|z_i| \leqslant \lambda$, then $prox_\sigma[F^*](z)_i = z_i$; if $|z_i| > \lambda$, then z_i' is chosen to be closest to z_i while respecting $|z_i'| \leqslant \lambda$ which leads to a scaling of the magnitude of z_i and $prox_\sigma[F^*](z)_i = \lambda z_i/|z_i|$. Note that the constant σ does not enter into this calculation. Putting the cases and components all together yields the second part of the proximal mapping in equation (45).

For the KL-TV problem, the proximal mapping for the data term is computed from equation (48):

$$prox_\sigma[F_1^*](p) = \arg\min_{p'} \left\{ \frac{\|p - p'\|_2^2}{2\sigma} - \sum_i [g \ln pos(\mathbf{1}_D - p')]_i + \delta_P(\mathbf{1}_D - p') \right\}.$$

We note that the objective is a smooth function in the positive orthant of $p' \in D$. Accordingly, we differentiate the objective with respect to p' ignoring the $pos(\cdot)$ and indicator functions, keeping in mind that we have to check that the minimizer p' is non-negative. Performing the differentiation and setting to zero yields the following quadratic equation:

$$p'^2 - (\mathbf{1}_D + p)p' + p - \sigma g = 0,$$

and substituting into the quadratic equation yields

$$p' = \tfrac{1}{2}(\mathbf{1}_D + p \pm \sqrt{(\mathbf{1}_D - p)^2 + 4\sigma g}).$$

We have two possible solutions, but it turns out that applying the restriction $\mathbf{1}_D - p' \geqslant 0$ selects the negative root. To see this, we evaluate $\mathbf{1}_D - p'$ at both roots:

$$\mathbf{1}_D - p' = \tfrac{1}{2}(\mathbf{1}_D - p \mp \sqrt{(\mathbf{1}_D - p)^2 + 4\sigma g}).$$

Using the fact that the data are non-negative, we have

$$\sqrt{(\mathbf{1}_D - p)^2 + 4\sigma g} \geqslant |\mathbf{1}_D - p|;$$

the positive root clearly leads to possible negative values for $\mathbf{1}_D - p'$ while the negative root respects $\mathbf{1}_D - p' \geqslant 0$ and yields equation (50).

For the final computation of a proximal mapping, we take a look at the data term of the constrained, TV-minimization problem. From equation (58), the proximal mapping of interest is evaluated by

$$prox_\sigma[F_1^*](p) = \arg\min_{p'} \left\{ \frac{\|p - p'\|_2^2}{2\sigma} + \epsilon\|p'\|_2 + \langle p', g\rangle_D \right\}.$$

Note the first term in the objective is spherically symmetric about p and increasing with distance from p, and the second term is also spherically symmetric about $\mathbf{0}_D$ and increasing with distance from $\mathbf{0}_D$. If just these two terms were present, the minimum would lie on the line segment between $\mathbf{0}_D$ and p. The third term, however, complicates the situation a little. We note that this term is linear in p', and it can be combined with the first term by completing the square. Performing this manipulation and ignoring constant terms (independent of p') yields

$$prox_\sigma[F_1^*](p) = \arg\min_{p'} \left\{ \frac{\|p - p' - \sigma g\|_2^2}{2\sigma} + \epsilon\|p'\|_2 \right\}.$$

By the geometric considerations discussed above, the minimizer lies on the line segment between $\mathbf{0}_D$ and $p - \sigma g$. Analyzing this one-dimensional minimization leads to equation (60).

Appendix D. The finite-differencing form of the image gradient and divergence

In this appendix, we write down the explicit forms of the finite differencing approximations of ∇ and $-\text{div}$ in two dimensions used in this paper. We use $x \in I$ to represent an $M \times M$ image and $x_{i,j}$ to refer to the (i, j)th pixel of x. To specify the linear transform ∇, we introduce the differencing images $\Delta_s x \in I$ and $\Delta_t x \in I$:

$$\Delta_s x_{i,j} = \begin{cases} x_{i+1,j} - x_{i,j} & i < M \\ -x_{i,j} & i = M, \end{cases}$$

$$\Delta_t x_{i,j} = \begin{cases} x_{i,j+1} - x_{i,j} & j < M \\ -x_{i,j} & j = M. \end{cases}$$

Using these definitions, ∇ can be written as

$$\nabla x = \begin{pmatrix} \Delta_s x \\ \Delta_t x \end{pmatrix}.$$

With this form of ∇, its transpose $-\mathrm{div}$ becomes

$$-\mathrm{div}\begin{pmatrix}\Delta_s x\\\Delta_t x\end{pmatrix}=\{-(\Delta_s x_{i,j}-\Delta_s x_{i-1,j})-(\Delta_t x_{i,j}-\Delta_t x_{i,j-1}),\, i \text{ and } j \in [1, M]\},$$

where the elements referred to outside the image border are set to zero: $\Delta_s x_{0,j}=\Delta_s x_{i,0}=\Delta_t x_{0,j}=\Delta_t x_{i,0}=0$. What the particular form of ∇ is in its discrete form is not that important, but it is critical that the discrete forms of $-div$ and ∇ are the transposes of each other.

Appendix E. Preconditioned CP algorithm demonstrated on the KL-TV optimization problem

Chambolle and Pock followed their article, Chambolle and Pock (2011), with a pre-conditioned version of their algorithm that suits our purpose of optimization problem prototyping while potentially improving algorithm efficiency substantially for the ℓ_2^2-TV and KL-TV optimization problems with small λ. The new algorithm replaces the constants σ and τ with vector quantities that are computed directly from the system matrix K, which yields a vector in space Y from a vector in space X. One form of the suggested, diagonal pre-conditioners uses the following weights:

$$\Sigma=\frac{\mathbf{1}_Y}{|K|\mathbf{1}_X}, \tag{E.1}$$

$$\mathrm{T}=\frac{\mathbf{1}_X}{|K|^T\mathbf{1}_Y}, \tag{E.2}$$

where $\Sigma \in Y$, $\mathrm{T} \in X$, and $|K|$ is the matrix formed by taking the absolute value of each element of K. In order to generate the CP algorithm instance incorporating pre-conditioning, the proximal mapping needs to be modified:

$$prox_\Sigma[F](y)=\arg\min_{y'}\left\{F(y')+\frac{1}{2}(y-y')^T\left(\frac{y-y'}{\Sigma}\right)\right\}. \tag{E.3}$$

The second term in this minimization is still quadratic but no longer spherically symmetric. The difficulty in deriving the pre-conditioned CP algorithm instances is similar to that of the original algorithm. On the one hand there is no need for finding $\|K\|_2$, but on the other hand deriving the proximal mapping may become more involved. For the ℓ_2^2-TV and the KL-TV optimization problems, the proximal mapping is simple to derive and it turns out that the mappings can be arrived at by replacing σ by Σ and τ by T.

The gain in efficiency for small λ comes from being able to absorb this parameter into the TV term and allowing Σ to account for the mismatch between TV and data agreement terms. We modify the definitions of ∇ and $-div$ matrices from appendix D:

$$\nabla_\lambda x=\begin{pmatrix}\lambda\Delta_s x\\\lambda\Delta_t x\end{pmatrix},$$

and

$$-\mathrm{div}_\lambda\begin{pmatrix}\Delta_s x\\\Delta_t x\end{pmatrix}=\{-\lambda(\Delta_s x_{i,j}-\Delta_s x_{i-1,j})-\lambda(\Delta_t x_{i,j}-\Delta_t x_{i,j-1}),\, i \text{ and } j \in [1, M]\},$$

where again the elements referred to outside the image border are set to zero: $\Delta_s x_{0,j}=\Delta_s x_{i,0}=\Delta_t x_{0,j}=\Delta_t x_{i,0}=0$.
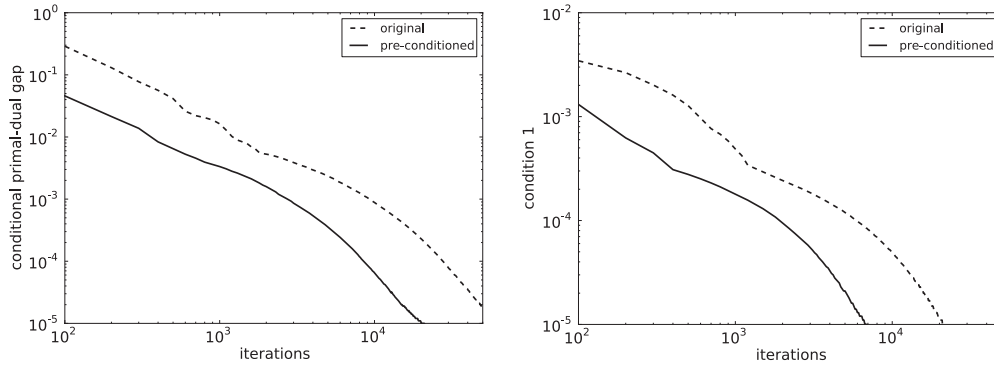
Figure E1. Left: convergence of the partial primal–dual gap for the CP algorithm instance solving equation (46) for $\lambda = 2 \times 10^{-5}$ for the original and pre-conditioned CP algorithm. Right: plot indicating agreement with condition 1 for the KL-TV optimization problem. See figure 3 for explanation.

For a complete example, we write the pre-conditioned CP algorithm instance for KL-TV in listing 9. To illustrate the potential gain in efficiency, we show the condition primal–dual gap as a function of iteration number for the KL-TV problem with $\lambda = 2 \times 10^{-5}$ in figure E1. While we have presented the pre-conditioned CP algorithm as a patch for the small λ case, it really provides an alternative prototyping algorithm and it can be used instead of the original CP algorithm.

Algorithm 9. Pseudocode for N steps of the KL-TV pre-conditioned CP algorithm instance.

1: $\Sigma_1 \leftarrow \mathbf{1}_D/(|A|\mathbf{1}_I)$; $\Sigma_2 \leftarrow \mathbf{1}_V/(|\nabla_\lambda|\mathbf{1}_I)$; $\mathrm{T} \leftarrow \mathbf{1}_I/(|A^T|\mathbf{1}_D + |div_\lambda|\mathbf{1}_V)$
2: $\theta \leftarrow 1$; $n \leftarrow 0$
3: initialize u_0, p_0, and q_0 to zero values
4: $\bar{u}_0 \leftarrow u_0$
5: **repeat**
6:   $p_{n+1} \leftarrow \frac{1}{2}\left(\mathbf{1}_D + p_n + \Sigma_1 A\bar{u}_n - \sqrt{(p_n + \Sigma_1 A\bar{u}_n - \mathbf{1}_D)^2 + 4\Sigma_1 g}\right)$
7:   $q_{n+1} \leftarrow (q_n + \Sigma_2\nabla_\lambda\bar{u}_n)/\max(\mathbf{1}_I, |q_n + \Sigma_2\nabla_\lambda\bar{u}_n|)$
8:   $u_{n+1} \leftarrow u_n - \mathrm{T}A^T p_{n+1} + \mathrm{T}div_\lambda\, q_{n+1}$
9:   $\bar{u}_{n+1} \leftarrow u_{n+1} + \theta(u_{n+1} - u_n)$
10:   $n \leftarrow n + 1$
11: **until** $n \geqslant N$

References

Barrett H H and Myers K J 2004 *Foundations of Image Science* (Hoboken, NJ: Wiley)

Beck A and Teboulle M 2009 Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems *IEEE Trans. Image Process.* **18** 2419–34

Becker S R, Candes E J and Grant M 2011 Templates for convex cone problems with applications to sparse signal recovery *Math. Prog. Comp.* **3** 165–218

Bian J, Siewerdsen J H, Han X, Sidky E Y, Prince J L, Pelizzari C A and Pan X 2010 Evaluation of sparse-view reconstruction from flat-panel-detector cone-beam CT *Phys. Med. Biol* **55** 6575–99

Boyd S P and Vandenberghe L 2004 *Convex Optimization* (Cambridge: Cambridge University Press)

Candès E J and Wakin M B 2008 An introduction to compressive sampling *IEEE Signal Process. Mag.* **25** 21–30

Chambolle A and Pock T 2011 A first-order primal–dual algorithm for convex problems with applications to imaging *J. Math. Imag. Vis.* **40** 1–26

Chan T F and Esedoglu S 2005 Aspects of total variation regularized L^1 function approximation *SIAM J. Appl. Math.* **65** 1817–37

Chen G H, Tang J and Leng S 2008 Prior image constrained compressed sensing (PICCS): a method to accurately reconstruct dynamic CT images from highly undersampled projection data sets *Med. Phys.* **35** 660–3

Choi K, Wang J, Zhu L, Suh T-S, Boyd S and Xing L 2010 Compressed sensing based cone-beam computed tomography reconstruction with a first-order method *Med. Phys.* **37** 5113–25

Combettes P L and Pesquet J C 2008 A proximal decomposition method for solving convex variational inverse problems *Inverse Problems* **24** 065014

De Man B and Basu S 2004 Distance-driven projection and backprojection in three dimensions *Phys. Med. Biol.* **49** 2463–75

Defrise M, Vanhove C and Liu X 2011 An algorithm for total variation regularization in high-dimensional linear problems *Inverse Problems* **27** 065002

Elad M 2010 *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing* (Berlin: Springer)

Erdogan H and Fessler J A 1999 Ordered subsets algorithms for transmission tomography *Phys. Med. Biol* **44** 2835–52

Green P 1984 Iteratively reweighted least squares for maximum likelihood estimation and some robust and resistant alternatives *J. R. Stat. Soc.* B **46** 149–92

Han X, Bian J, Eaker D R, Kline T L, Sidky E Y, Ritman E L and Pan X 2011 Algorithm-enabled low-dose micro-CT imaging *IEEE Trans. Med. Imag.* **30** 606–20

Jensen T L, Jørgensen J H, Hansen P C and Jensen S H 2011 Implementation of an optimal first-order method for strongly convex total variation regularization *BIT* at press (online http://www.springerlink.com/index/10.1007/s10543-011-0359-8)

Jørgensen J H, Sidky E Y and Pan X 2011a Analysis of discrete-to-discrete imaging models for iterative tomographic image reconstruction and compressive sensing *IEEE Trans. Med. Imag.* (arXiv:1109.0629) submitted

Jørgensen J H, Hansen P C, Sidky E Y, Reiser I S and Pan X 2011b Toward optimal x-ray flux utilization in breast CT *Proc. 11th Int. Meeting on Three-Dimensional Image Reconstruction in Radiology and Nuclear Medicine (Potsdam, Germany, 11–15 July 2011)* pp 359–62 (arXiv:1104.1588)

Li M, Yang H and Kudo H 2002 An accurate iterative reconstruction algorithm for sparse objects: application to 3D blood vessel reconstruction from a limited number of projections *Phys. Med. Biol.* **47** 2599–609

McCollough C H, Primak A N, Braun N, Kofler J, Yu L and Christner J 2009 Strategies for reducing radiation dose in CT *Radiol. Clin. N. Am.* **47** 27–40

Nocedal J and Wright S 2006 *Numerical Optimization* 2nd edn (Berlin: Springer)

Pan X, Sidky E Y and Vannier M 2009 Why do commercial CT scanners still employ traditional, filtered back-projection for image reconstruction? *Inverse Problems* **25** 123009

Pock T and Chambolle A 2011 Diagonal preconditioning for first order primal–dual algorithms in convex optimization *IEEE Int. Conf. Computer Vision (ICCV 2011)* pp 1762–69

Ramani S and Fessler J 2011 A splitting-based iterative algorithm for accelerated statistical x-ray CT reconstruction *IEEE Trans. Med. Imag.* **31** 677–88

Reiser I and Nishikawa R M 2010 Task-based assessment of breast tomosynthesis: effect of acquisition parameters and quantum noise *Med. Phys.* **37** 1591–600

Ritschl L, Bergner F, Fleischmann C and Kachelrieß M 2011 Improved total variation-based CT image reconstruction applied to clinical data *Phys. Med. Biol.* **56** 1545–62

Rockafellar R T 1970 *Convex Analysis* (Princeton, NJ: Princeton University Press)

Siddon R L 1985 Fast calculation of the exact radiological path for a three-dimensional CT array *Med. Phys.* **12** 252–5

Sidky E Y, Anastasio M A and Pan X 2010 Image reconstruction exploiting object sparsity in boundary-enhanced x-ray phase-contrast tomography *Opt. Express* **18** 10404–22

Sidky E Y, Duchin Y, Ullberg C and Pan X 2011 X-ray computed tomography: advances in image formation. A constrained, total-variation minimization algorithm for low-intensity x-ray CT *Med. Phys.* **38** S117–25

Sidky E Y, Kao C-M and Pan X 2006 Accurate image reconstruction from few-views and limited-angle data in divergent-beam CT *J. X-Ray Sci. Tech.* **14** 119–39

Sidky E Y and Pan X 2008 Image reconstruction in circular cone-beam computed tomography by constrained, total-variation minimization *Phys. Med. Biol.* **53** 4777–807

Xia D, Xiao X, Bian J, Han X, Sidky E Y, Carlo F D and Pan X 2011 Image reconstruction from sparse data in synchrotron-radiation-based microtomography *Rev. Sci. Instrum.* **82** 043706

Xu F and Mueller K 2007 Real-time 3D computed tomographic reconstruction using commodity graphics hardware *Phys. Med. Biol.* **52** 3405–19

Yin W, Osher S, Goldfarb D and Darbon J 2008 Bregman iterative algorithms for ℓ1-minimization with applications to compressed sensing *SIAM J. Imag. Sci.* **1** 143–68

Zeng G L and Gullberg G T 2000 Unmatched projector/backprojector pairs in an iterative reconstruction algorithm *IEEE Trans. Med. Imag.* **19** 548–55

Ziegler A, Nielsen T and Grass M 2008 Iterative reconstruction of a region of interest for transmission tomography *Med. Phys.* **35** 1317–27