LBL--32965

DE93 004694

Generalized Approach to Inverse Problems in Tomography: Image Reconstruction for Spatially Variant Systems Using Natural Pixels

John R. Baker[†‡], Thomas F. Budinger[†‡], and Ronald H. Huesman[‡]
[†] Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
and
[‡] Research Medicine and Radiation Biophysics Division
Lawrence Berkeley Laboratory
University of California
1 Cyclotron Road
Berkeley, CA 94720

[1]This work was supported in part by the Director, Office of Energy Research, Office of Health and Environmental Research, Medical Applications and Biophysical Research Division of the U.S. Department of Energy under contract No DE-AC03-SF00098 and in part by NIH grant HL07367.

Abstract

A major limitation in tomographic inverse problems is inadequate computation speed, which frequently impedes the application of engineering ideas and principles in medical science more than in the physical and engineering sciences. Medical problems are computationally taxing because a minimum description of the system often involves 5 dimensions (3 space, 1 energy, 1 time), with the range of each space coordinate requiring up to 512 samples. The computational tasks for this problem can be simply expressed by posing the problem as one in which the tomograph system response function is spatially invariant, and the noise is additive and Gaussian. Under these assumptions, a number of reconstruction methods have been implemented with generally satisfactory results for general medical imaging purposes. However, if the system response function of the tomograph is assumed more realistically to be spatially variant and the noise to be Poisson, the computational problem becomes much more difficult. Some of the algorithms being studied to compensate for position dependent resolution and statistical fluctuations in the data acquisition process, when expressed in canonical form, are not practical for clinical applications because the number of computations necessary exceeds the capabilities of high performance computer systems currently available. Reconstruction methods based on *natural pixels*, specifically orthonormal natural pixels, preserve symmetries in the data acquisition process. Fast implementations of orthonormal natural pixel algorithms can achieve orders of magnitude speedup relative to general implementations. Thus, specialized thought in algorithm development can lead to more significant increases in performance than can be achieved through hardware improvements alone.

1 Background

The tomographic projection formation can be described by the discrete-continuous model [1], [2], [3], [4]

$$p_{\theta k} = F_{\theta k} \cdot b \tag{1}$$

$$= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx \, f_{\theta k}(x,y) \, b(x,y) \tag{2}$$

where $p_{\theta k}$ is the measured projection at angle index θ and bin position k. $F_{\theta k}$ is a second order tensor functional operating on the two dimensional object distribution b. The operation of equation 1 represents the integration of the product of the impulse response $f_{\theta k}(x,y)$ and the object distribution $b(x,y)$ over the imaging field as depicted in figure 1. The symbol $\cdot$ indicates integration over the imaging field. The impulse response function is the spatial response of a projection at angle θ and bin k to a point source moved to every position within the sampling domain. There are Θ different angles and K projection bins at each angle.

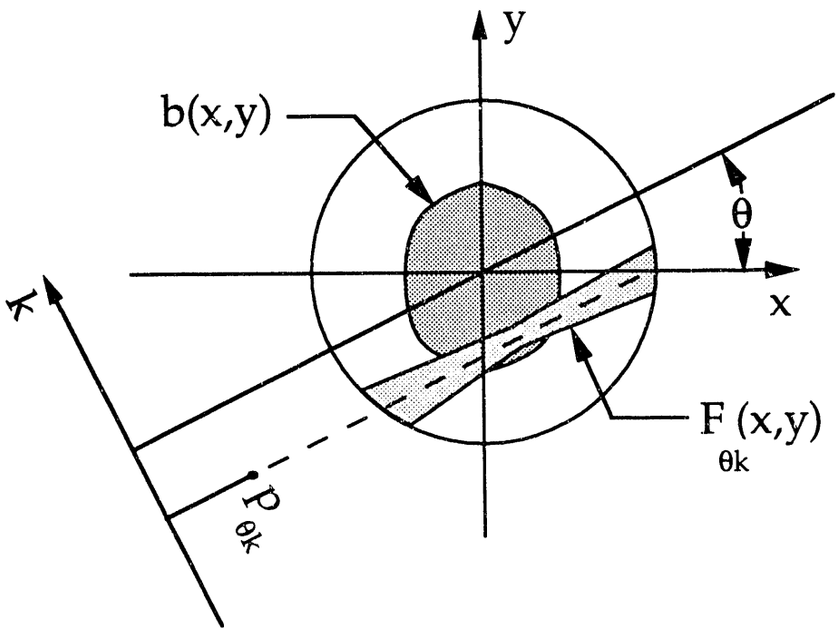

Figure 1: Schematic of projection formation.

To simplify notation[1], the projection formation equation is written in vector form

$$\boldsymbol{p} = F \cdot b \tag{3}$$

[1]Lower case bold symbols denote vectors, lower case script symbols denote functions or scalars, upper case bold symbols denote matrices, and upper case script symbols denote operators.

by combining the θ and k indices into one index. Specifically[5],

$$b \in L_2\left[\mathbf{R}^2\right], \tag{4}$$

$$\boldsymbol{p} \in \mathbf{R}^{\Theta K}, \text{ and} \tag{5}$$

$$F : L_2\left[\mathbf{R}^2\right] \mapsto \mathbf{R}^{\Theta K}. \tag{6}$$

Because the model is based on the fact that the detection process is defined on a discrete domain and the original distribution is defined on a continuous domain, the model is easily adapted to include a variety of physical effects found in many imaging modalities. For positron emission tomography (PET), $F_{\theta k}$ can include radioactive decay, positron range, non-collinearity of photons, sampling geometry, attenuation, inter-detector-crystal scatter, detector crystal penetration, and detection efficiency [6] [7] [8].

The singular value decomposition of the tomograph system response function is

$$F = \boldsymbol{U} S \cdot V^T. \tag{7}$$

$\boldsymbol{U}$ is an orthogonal matrix containing the left singular vectors of F and is defined by the eigenvalue decomposition of the projection normal matrix,

$$\boldsymbol{A} \equiv F \cdot F^T \tag{8}$$

$$= \boldsymbol{U} S \cdot S^T \boldsymbol{U}^T \tag{9}$$

where F^T is the adjoint operator of F [5]. An element $A_{\theta' k' \theta k}$ is the projection at angle θ' and bin k' of the backprojection at angle θ and bin k of a unit projection value, $p_{\theta k} = 1$. A schematic of this operation is shown in figure 2. The projection normal matrix is symmetric and positive semidefinite. V are the right singular functions of F and are defined by the relationship

$$F^T F = V \cdot S^T S \cdot V^T. \tag{10}$$

The functional S operates similarly to F in that it maps continuous domain functions to discrete domain samples. Thus, V performs an infinite dimensional rotation on the continuous domain object space, S selects and scales a finite number of the rotated functions, and $\boldsymbol{U}$ performs a finite dimensional rotation into the discrete domain projection measurement space as shown in equations 11-13.

$$\boldsymbol{U} : \mathbf{R}^{\Theta K} \mapsto \mathbf{R}^{\Theta K}, \tag{11}$$

$$S : L_2\left[\mathbf{R}^2\right] \mapsto \mathbf{R}^{\Theta K}, \text{ and} \tag{12}$$

$$V : L_2\left[\mathbf{R}^2\right] \mapsto L_2\left[\mathbf{R}^2\right]. \tag{13}$$

A geometric interpretation is given in figure 3.

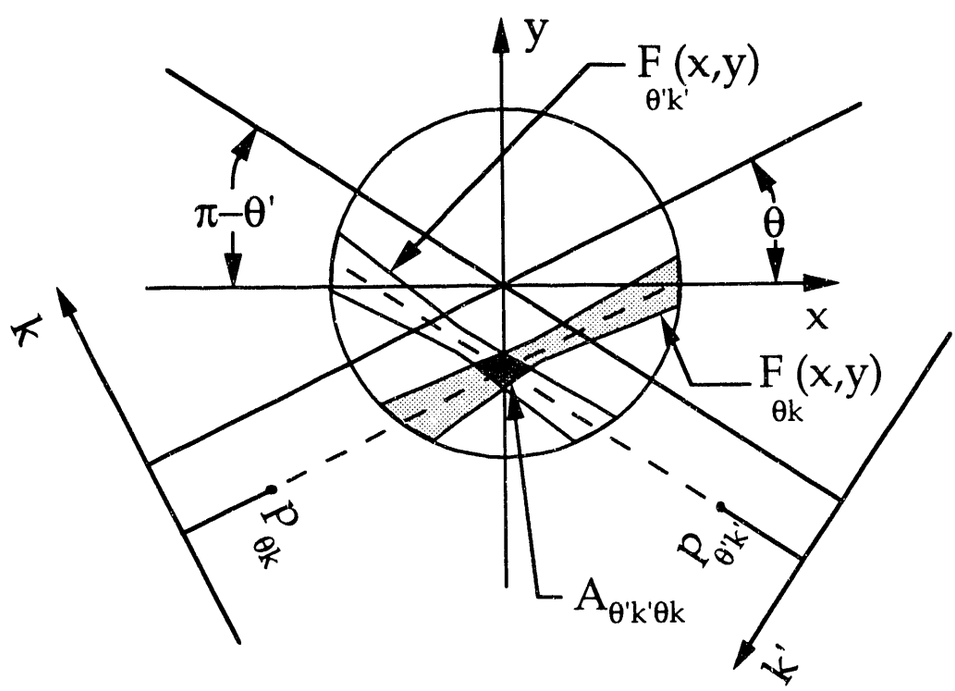

Figure 2: Schematic of the computation of one element of the projection normal matrix. A unit projection is backprojected and reprojected to a new projection to form a projection matrix element.

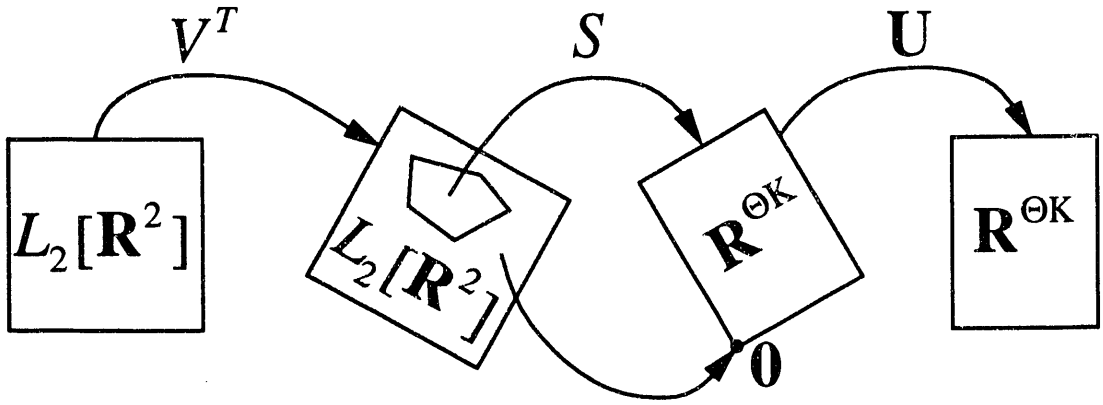

Figure 3: Singular value decomposition of the projection operator. The operator V^T performs an infinite dimensional rotation on the continuous domain object space, S selects and scales a finite number of the rotated functions, and $\boldsymbol{U}$ performs a finite dimensional rotation into the discrete domain projection measurement space.

The goal of tomography is to reconstruct the unknown distribution, b, from one realization, $\underline{\boldsymbol{p}}$, of the projection measurement process $\boldsymbol{p}$ [9] [10] [11] [12] [13] [14] [15] [16]. Reconstructing the true continuous space distribution, b, from sampled projections is probably impossible without prior information about the distribution [3] [17] [18] [19]. Instead, a discretized representation, c_{mn}, is estimated from the measurements where $B_{mn}(x,y)$ defines a generalized pixel.

$$b(x,y) \approx \sum_{mn} B^T_{mn}(x,y)c_{mn} \tag{14}$$

In vector form the pixelization is

$$b \approx B^T\boldsymbol{c}. \tag{15}$$

Using this representation, a least squares estimator (LSE) [5] [20] [21] [22] [23] is formulated to estimate the mean intensity of the generalized pixels, $\hat{\boldsymbol{c}}$, used to describe the unknown spatial distribution from one measured projection dataset, $\underline{\boldsymbol{p}}$. While the resulting LSE formula is quite general, specific application to image reconstruction using square pixels, Buonocore's natural pixels [24] [25], and the new orthonormal natural pixels is shown. The least squares estimator for the mean intensity of the generalized pixel image is found by minimizing the square of the L_2 norm of the difference between the projection vector, $\boldsymbol{p}$, and the estimated projection, $F \cdot B^T\boldsymbol{c}$, over all possible image vectors, $\boldsymbol{c}$; i.e.,

$$\hat{\boldsymbol{c}} \equiv \arg\min_{\boldsymbol{c}} \left\{ \left\| \boldsymbol{p} - F \cdot B^T \boldsymbol{c} \right\|_2^2 \right\} \tag{16}$$

$$= \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \boldsymbol{p}. \tag{17}$$

The operator $^+$ is the Moore-Penrose pseudo-inverse [26] [27] [28]. Using the measured projection vector, $\underline{\boldsymbol{p}}$, as a single sample estimate of the mean projection vector gives

$$\hat{\boldsymbol{c}} = \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{18}$$

The fluctuations of the generalized pixel least squares estimator due to random variations in the measurements can typically be characterized in terms of the covariance between pixel estimates. The covariance matrix for the generalized pixel estimator is

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} \equiv E\left[\left(\hat{\mathbf{c}} - E\hat{\mathbf{c}} \right) \left(\hat{\mathbf{c}} - E\hat{\mathbf{c}} \right)^T \right] \tag{19}$$

$$= \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U} \left(B \cdot V \cdot S^T \right)^+ . \tag{20}$$

where $\boldsymbol{\Sigma}_{\boldsymbol{p}}$ is the covariance of the projections.

The estimator for the mean of the intensity of the object in continuous space is found by applying the adjoint of the basis operator, B, to the generalized pixel estimator of the intensity mean. For the continuous space object, the least squares estimator for the mean of the intensity is

$$\hat{b} = B^T \hat{\mathbf{c}} \tag{21}$$

$$= B^T \left(S \cdot V^T \cdot B^T \right)^+ \boldsymbol{U}^T \boldsymbol{p}. \tag{22}$$

The basis set used to describe the pixels influences the types of artifacts that appear in the reconstructed image [25] [29] [30]. In this work, three bases are evaluated. The first, $B1$, is the traditional square pixel or Heaviside basis. The second basis set, $B2$, consists of the set of functions that comprise the tomograph system response functional and was proposed by Buonocore [24]. The third, $B3$, has been proposed by us and is composed of the right singular functions V, defined by equation 7, that have been selected by S and normalized by the L_2 norm.

$$B1 \equiv \text{Heaviside} \tag{23}$$

$$B2 \equiv F \tag{24}$$

$$B3 \equiv \left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}} S \cdot V^T \tag{25}$$

Small singular values in the singular value filter, $\left[\left(S \cdot S^T \right)^+ \right]^{\frac{1}{2}}$, of equation 18 can lead to large statistical errors in the reconstructed image, $\hat{b}$. By applying a diagonal weighting

matrix, $\boldsymbol{D}$, to the filter, the mean square error of the object estimates may be decreased [3]. The resulting estimates and covariance are

$$\hat{\boldsymbol{c}} = \boldsymbol{D}\left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{26}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \boldsymbol{D}\left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{U}^T \boldsymbol{\Sigma}_{\boldsymbol{p}} \boldsymbol{U}\left[\left(S \cdot S^T\right)^+\right]^{\frac{1}{2}} \boldsymbol{D} \tag{27}$$

$$\hat{b} = V \cdot S^T \boldsymbol{D}\left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{28}$$

Determining the weighting values is the subject of ongoing research. Since basis vectors are assumed to be arranged so the singular values are in the accepted non-increasing order, one possibility is to truncate the number of singular values used in the singular value filter so only the J largest singular values will be included. A weighting matrix with elements

$$D_{j'j} = \begin{cases} 1 & \text{if } j' = j \text{ and } j < J, \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

will select only the J largest singular values. Since the basis is orthonormal, the resulting object estimate is the sum of the estimates of each pixel that was multiplied by one; i.e.,

$$\hat{b} = \sum_{j=0}^{J-1} B3_j^T \hat{c}_j \tag{30}$$

$$= \sum_{j=0}^{J-1} V \cdot S_j^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{31}$$

2 Example

As a simple example, consider the spatial sampling system of figure 4 that has three projection angles and two projection bins at each angle. The angles are equally spaced between zero and π radians. The impulse response functions are defined by

$$f_{\theta k}(x,y) = \begin{cases} 1 & \text{if } k-1 \le -x\sin\left(\theta\frac{\pi}{3}\right) + y\cos\left(\theta\frac{\pi}{3}\right) < k \text{ and } x^2+y^2 \le 1, \\ 0 & \text{otherwise} \end{cases} \tag{32}$$

and are shown in figure 4. The projection normal matrix and the singular value decomposition of the projection normal matrix for this example are

$$
\begin{aligned}
\boldsymbol{A} &= F \cdot F^T \\
&= \frac{\pi}{6}\begin{bmatrix} 3 & 0 & 2 & 1 & 1 & 2 \\ 0 & 3 & 1 & 2 & 2 & 1 \\ 2 & 1 & 3 & 0 & 2 & 1 \\ 1 & 2 & 0 & 3 & 1 & 2 \\ 1 & 2 & 2 & 1 & 3 & 0 \\ 2 & 1 & 1 & 2 & 0 & 3 \end{bmatrix} \\
&= \boldsymbol{U}\boldsymbol{S} \cdot \boldsymbol{S}^T\boldsymbol{U}^T \\
&= \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -1 & -1 & 1 & 1 \\ 1 & 1 & -1 & -1 & 0 & -2 \\ 1 & -1 & 1 & 1 & 0 & -2 \\ 1 & 0 & -2 & 1 & -1 & 1 \\ 1 & 0 & 2 & -1 & -1 & 1 \end{bmatrix}
\begin{bmatrix} \frac{1}{\sqrt{6}} & & & & & \\ & \frac{1}{\sqrt{4}} & & & & \\ & & \frac{1}{\sqrt{12}} & & & \\ & & & \frac{1}{\sqrt{6}} & & \\ & & & & \frac{1}{\sqrt{4}} & \\ & & & & & \frac{1}{\sqrt{12}} \end{bmatrix} \\
&\quad \begin{bmatrix} \frac{9}{6\pi} & & & & & \\ & \frac{4}{6\pi} & & & & \\ & & \frac{4}{6\pi} & & & \\ & & & \frac{1}{6\pi} & & \\ & & & & 0 & \\ & & & & & 0 \end{bmatrix} \\
&\quad \begin{bmatrix} \frac{1}{\sqrt{6}} & & & & & \\ & \frac{1}{\sqrt{4}} & & & & \\ & & \frac{1}{\sqrt{12}} & & & \\ & & & \frac{1}{\sqrt{6}} & & \\ & & & & \frac{1}{\sqrt{4}} & \\ & & & & & \frac{1}{\sqrt{12}} \end{bmatrix}
\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & -2 & 2 \\ 1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 1 & 0 & 0 & -1 & -1 \\ 1 & 1 & -2 & -2 & 1 & 1 \end{bmatrix}
\end{aligned}
\tag{33, 34}
$$

where the left singular vectors, $\boldsymbol{U}$, are given as the product of a matrix and the inverse of the norm of that matrix. There are four non-zero singular values for this example parallel beam sampling system meaning that of the six measurements only four are linearly independent. The linear dependence between the measurements is simply demonstrated by subtracting one of the impulse response functions at one angle from all of the impulse response functions at the other angle, the resulting difference is equal to the impulse response function that was not included in the difference; e.g., for the system of figure 4,

$$f_{00}(\cdot,\cdot) + f_{01}(\cdot,\cdot) - f_{10}(\cdot,\cdot) = f_{11}(\cdot,\cdot). \tag{35}$$

Figures 5-7 show, respectively, a possible set of basis functions for square pixels, Buonocore's natural pixels, and orthonormal natural pixels using the sampling defined by the impulse response functions of figure 4.

Consider the wedge shaped object of figure 8 which is defined as

$$b(x,y) = \begin{cases} 1 & \text{if } \left|\frac{y}{x}\right| < \frac{1}{2} \text{ and } x \geq 0 \text{ and } x^2 + y^2 \leq 1, \\ 0 & \text{otherwise.} \end{cases} \tag{36}$$

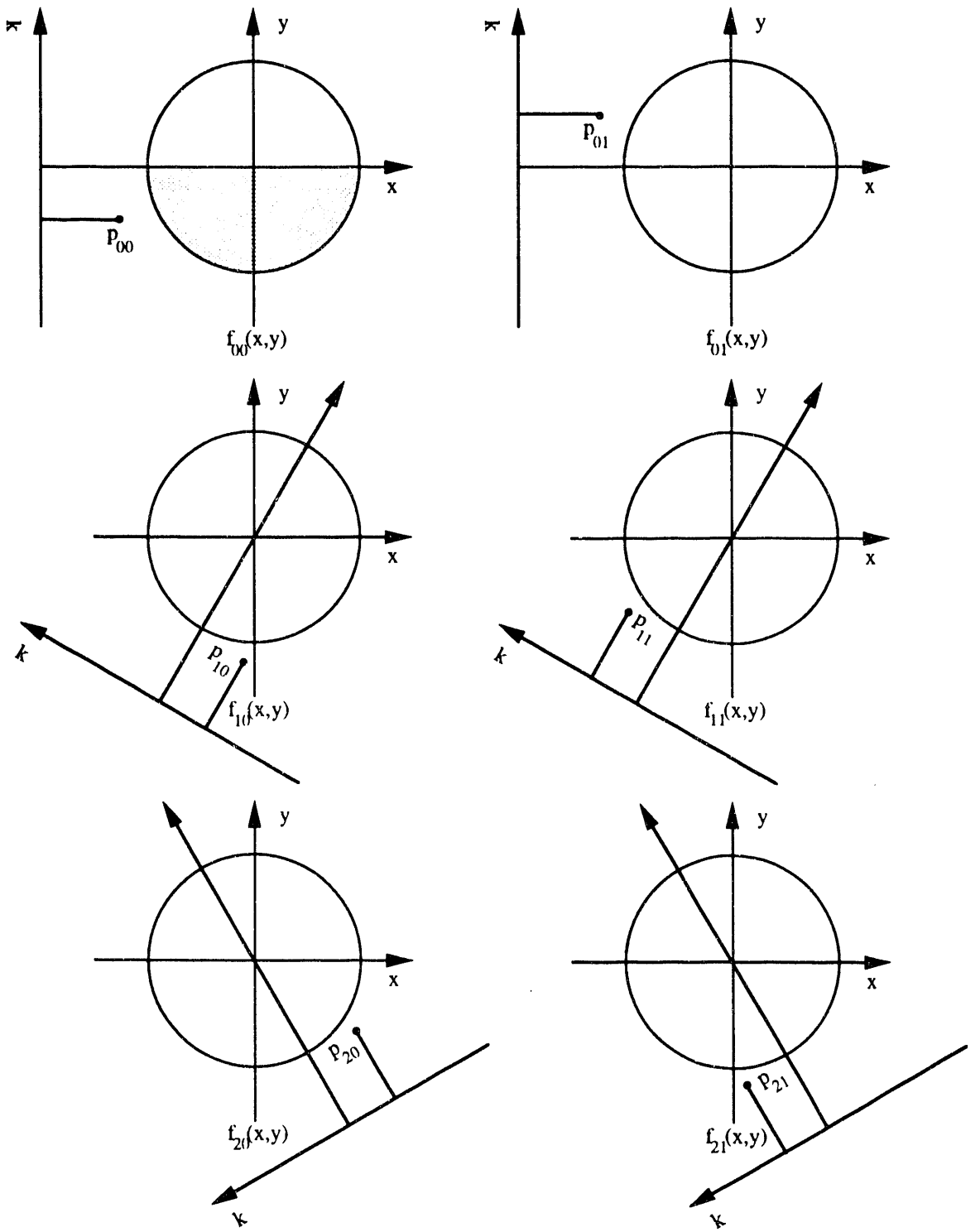

Figure 4: Schematic of spatial sampling functions for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

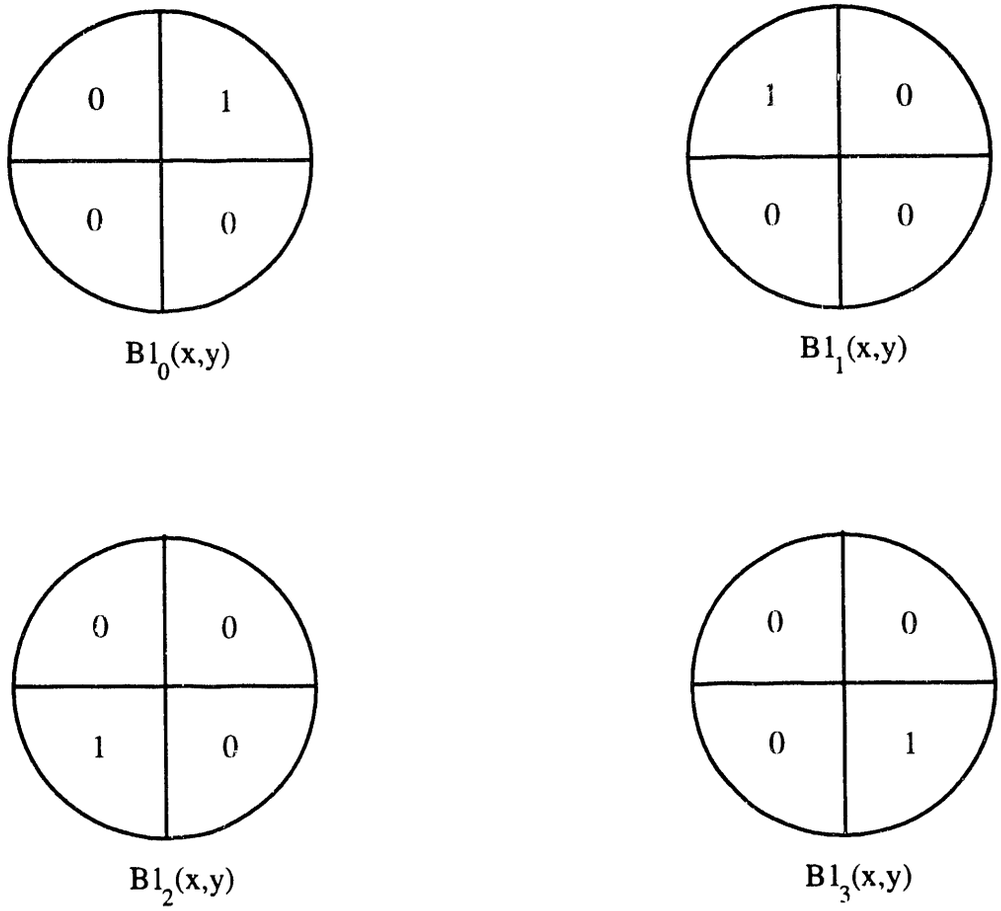

Figure 5: Square pixel or Heaviside basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

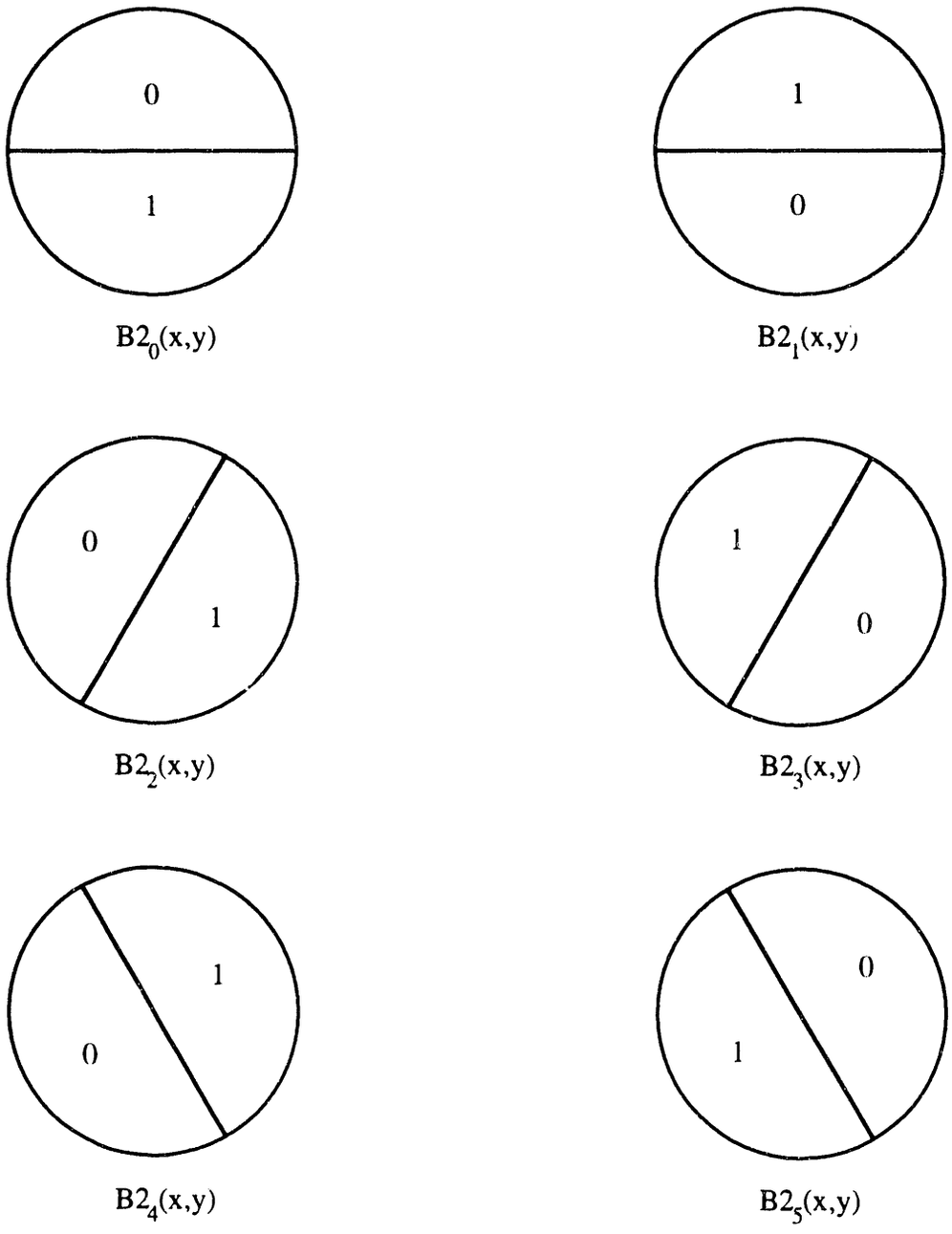

Figure 6: Buonocore's natural pixel basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

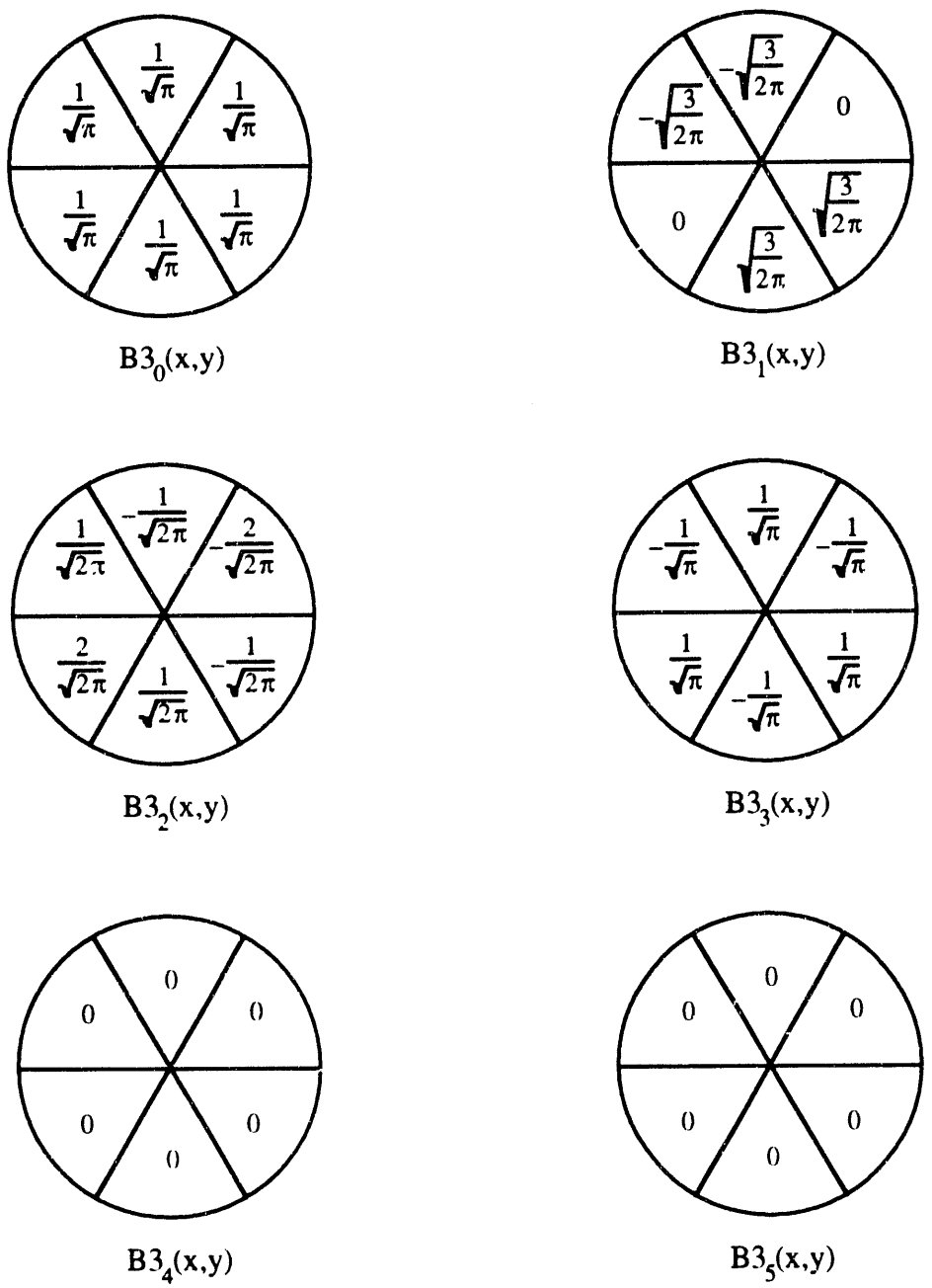

Figure 7: New orthonormal pixel basis for a simple parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

The pixel coefficients using the square pixel basis of figure 5 are

$$\boldsymbol{c} = \left[\tfrac{1}{3} \;\; 0 \;\; 0 \;\; \tfrac{1}{3} \right]^T. \tag{37}$$

In Buonocore's natural pixel basis of figure 6, the pixel coefficients are

$$\boldsymbol{c} = \frac{1}{72} \left[4 \;\; 4 \;\; 13 \;\; -5 \;\; 13 \;\; -5 \right]^T \tag{38}$$

and for the orthonormal natural pixel basis shown in figure 7

$$\boldsymbol{c} = \frac{\sqrt{\pi}}{4} \left[1 \;\; \tfrac{1}{\sqrt{6}} \;\; -\tfrac{1}{\sqrt{2}} \;\; 0 \;\; 0 \;\; 0 \right]^T. \tag{39}$$

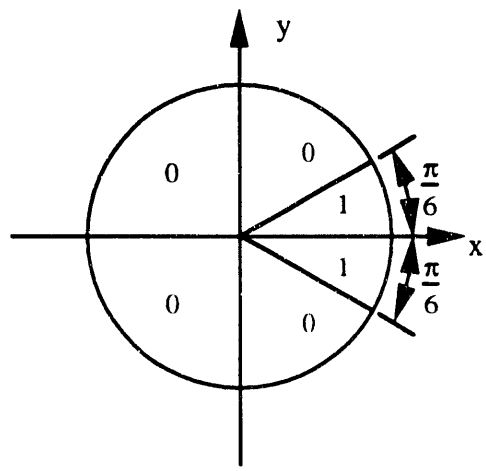

Figure 8: Wedge shaped phantom object.

Figure 9 shows, respectively, the continuous space representation of the wedge shaped object using square pixels, Buonocore's natural pixels, and orthonormal natural pixels with the sampling defined by the impulse response functions of figure 4.

All systematic reprojection errors due to pixelization are eliminated using a basis like $B2$ or $B3$ as shown in table 1 for the wedge phantom. While this is true in particular for $B2$ and $B3$, any basis that spans the subspace of functions defined by $S \cdot V^T$ will also have this property. A geometric interpretation of the error associated with computing projections of pixelized object is shown in figure 10. Many of these bases may represent the original object distribution, b, better than $B2$ or $B3$. However, no information about the coefficients for the functions that are outside the space $S \cdot V^T$ is available from the projection measurements. By using *a priori* information about the continuous space distribution of b, the formulation of Bayesian estimators that use basis functions not in $S \cdot V^T$ is an exciting area for future

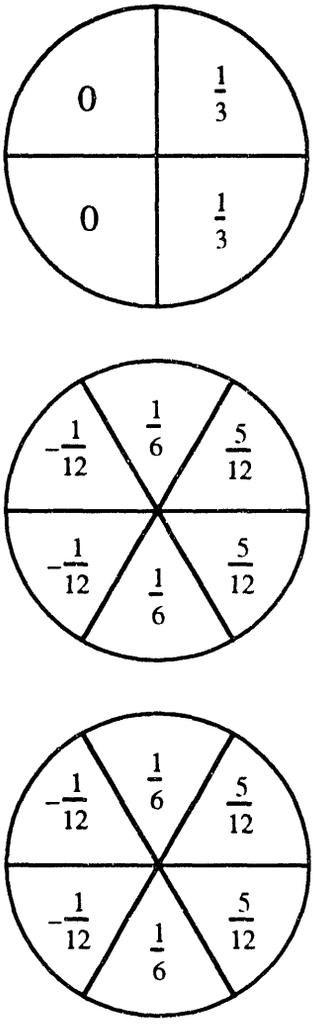

Figure 9: Representation of a wedge shaped object using top) square pixels, middle) Buonocore's natural pixels, and bottom) orthonormal natural pixels.

Basis	Object Estimate Squared Error $\left\|b - B^T\boldsymbol{c}\right\|_2^2$	Projection Estimate Squared Error $\left\|\boldsymbol{p} - F \cdot B^T\boldsymbol{c}\right\|_2^2$
$B1$	$\frac{4}{3}$	$\frac{\pi^2}{324}$
$B2$	$\frac{7}{6}$	0
$B3$	$\frac{7}{6}$	0

Table 1: Errors due to pixelization for a wedge shaped phantom sampled with a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

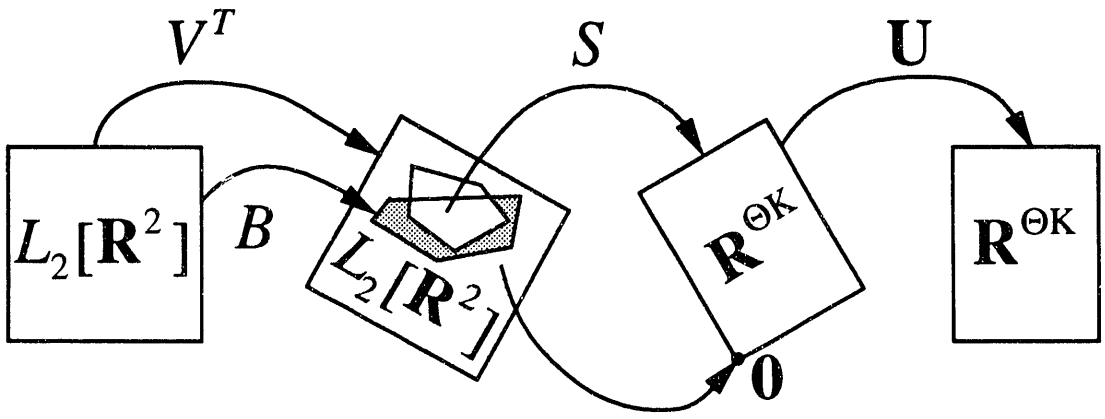

Figure 10: Simplified geometric representation of the systematic error that results from computing model projections from pixelized versions of an object. Pixel bases that include the subspace $S \cdot V^T$ eliminate systematic pixelization error.

research. Thus, the basis subset contained in $S \cdot V^T$ is from a channel model for F and the subset contained in $V^T - S \cdot V^T$ is from a process model for b.

To test the LSE in the presence of noise, a noisy projection dataset,

$$\boldsymbol{p} = \begin{bmatrix} 0.8579 & 0.2920 & 1.4252 & 0.0000 & 0.4829 & 0.0000 \end{bmatrix}^T \tag{40}$$

will be used. This projection dataset was created by sampling an independent multivariate normal (Gaussian) distribution with mean and variance equal to the noiseless projections of the wedge shaped object. The estimated pixel coefficients and the covariance of those estimates using the square pixel basis of figure 5 are

$$\hat{\boldsymbol{c}} = \begin{bmatrix} 0.5508 & -0.3602 & 0.0981 & 1.0092 \end{bmatrix}^T \tag{41}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \begin{bmatrix} 0.1738 & 0.0014 & -0.0854 & 0.0870 \\ 0.0014 & 0.0677 & -0.0191 & -0.0854 \\ -0.0854 & -0.0191 & 0.0677 & 0.0014 \\ 0.0870 & -0.0854 & 0.0014 & 0.1738 \end{bmatrix}. \tag{42}$$

In Buonocore's natural pixel basis of figure 6, the pixel coefficient estimates and covariance are

$$\hat{\boldsymbol{c}} = \begin{bmatrix} 0.1534 & 0.0629 & 0.5383 & -0.3220 & 0.1336 & 0.0827 \end{bmatrix}^T \tag{43}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \begin{bmatrix} 0.1810 & -0.1771 & -0.1428 & 0.1468 & 0.1556 & -0.1517 \\ -0.1771 & 0.1810 & 0.1556 & -0.1517 & -0.1428 & 0.1468 \\ -0.1428 & 0.1556 & 0.1899 & -0.1771 & -0.1384 & 0.1512 \\ 0.1468 & -0.1517 & -0.1771 & 0.1722 & 0.1512 & -0.1561 \\ 0.1556 & -0.1428 & -0.1384 & 0.1512 & 0.1899 & -0.1771 \\ -0.1517 & 0.1468 & 0.1512 & -0.1561 & -0.1771 & 0.1722 \end{bmatrix} \tag{44}$$

and for the orthonormal natural pixel basis shown in figure 7

$$\hat{\boldsymbol{c}} = \begin{bmatrix} 0.5751 & 0.6879 & -0.3641 & -0.2124 & 0.0000 & 0.0000 \end{bmatrix}^T \tag{45}$$

$$\boldsymbol{\Sigma}_{\hat{\boldsymbol{c}}} = \begin{bmatrix} 0.0556 & 0.0340 & -0.0589 & 0.0000 & 0.0000 & 0.0000 \\ 0.0340 & 0.1250 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ -0.0589 & 0.0000 & 0.1250 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.5000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \\ 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 & 0.0000 \end{bmatrix}. \tag{46}$$

Figure 11 shows, respectively, the continuous space reconstruction of the wedge shaped object using square pixels, Buonocore's natural pixels, and orthonormal natural pixels with the sampling defined by the impulse response functions of figure 4. Table 2 shows the mean and observed object squared error and the the observed projection squared error for these

reconstructions. The natural pixel and orthonormal natural pixel estimates have better observed squared error than the square pixel estimates. The square pixel estimator has better mean squared error characteristics than the unfiltered estimators based on natural pixels or orthonormal natural pixels for the wedge shaped object.

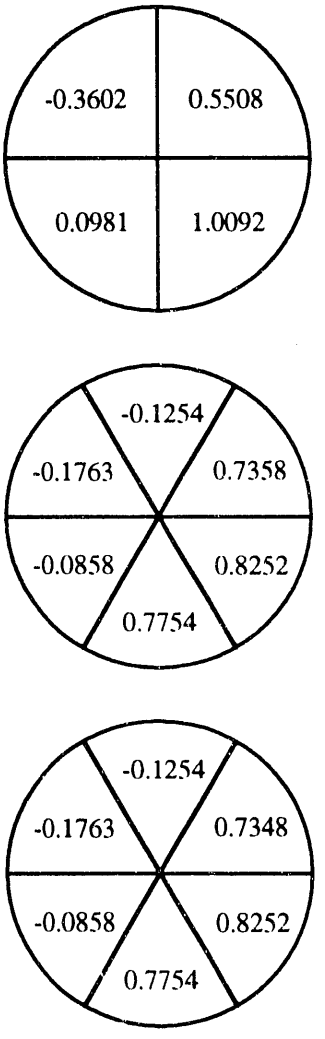

Figure 11: Least squares reconstruction of a wedge shaped object using top) square pixels, middle) Buonocore's natural pixels, and bottom) orthonormal natural pixels.

Table 3 shows the effects of using the diagonal weighting matrix defined in equation 29 with the value J varied from one to four for the example of figure 8. The projection estimate error decreases with the inclusion of each orthonormal natural pixel basis function; however, adding the image corresponding to the third basis vector, $J = 4$, increases the mean object squared error. The increase in mean object squared error is due to noise being added to the reconstructed image while no new information about the object is being added since the wedge phantom only has non-zero projections onto the first three orthonormal natural basis

Basis	Mean Object Squared Error $E\left\|b - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Object Squared Error $\left\|Eb - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Projection Squared Error $\left\|E\boldsymbol{p} - F \cdot B^T\hat{\mathbf{c}}\right\|_2^2$
$B1$	1.1775	3.2636	0.3001
$B2$	1.3571	2.6294	0.2348
$B3$	1.3571	2.6294	0.2348

Table 2: Least squares reconstruction errors for a wedge shaped phantom sampled with noise by a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

vectors as shown in equation 39.

Truncation Index J	Mean Object Squared Error $E\left\|b - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Object Squared Error $\left\|Eb - B^T\hat{\mathbf{c}}\right\|_2^2$	Observed Projection Squared Error $\left\|E\boldsymbol{p} - F \cdot B^T\hat{\mathbf{c}}\right\|_2^2$
1	1.4762	1.9655	1.5270
2	1.3839	2.8223	0.5360
3	0.8571	2.4571	0.2584
4	1.3571	2.6294	0.2348

Table 3: Effects of truncating the number of singular values included in the singular value filter on orthonormal natural pixel least squares reconstruction errors for a wedge shaped phantom sampled with noise by a parallel beam tomographic system with three equally spaced projection angles and two projection bins at each angle.

3 Implementation

The computational complexity of reconstruction algorithms can be reduced by using the orthonormal natural pixel basis because it preserves symmetries in the data acquisition process. In this section, an implementation of the orthonormal natural pixel least squares estimator for rotationally invariant systems is described and analyzed.

3.1 Orthonormal Least Squares

The continuous space representation of the truncated minimum L_2 norm least squares estimator for the mean value of the orthonormal natural pixel image is from equation 31

$$\hat{b} = \sum_{j=0}^{J-1} V \cdot S_j^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{47}$$

This estimate of the object is converted to a square pixel representation using the Heaviside basis operator, $B1$, of equation 23. The resulting square pixel representation is

$$\hat{c} = \sum_{j=0}^{J-1} B1 \cdot V \cdot S_j^T \left(S \cdot S^T\right)^+ \boldsymbol{U}^T \underline{\boldsymbol{p}} \tag{48}$$

and after rearrangement using equation 7,

$$\hat{c} = \sum_{j=0}^{J-1} B1 \cdot F^T \boldsymbol{U} \left(S \cdot S^T\right)_{jj}^+ \boldsymbol{U}^T \underline{\boldsymbol{p}}. \tag{49}$$

A block diagram of the orthonormal natural pixel least squares estimation algorithm is shown in figure 12. In general, the computation of the projection normal matrix requires $\Theta(\Theta^2 K^2)$ integral evaluations[2] [3]. An $\Omega(\Theta^3 K^3)$ singular value or eigenvalue decomposition routine is used to compute $\boldsymbol{U}$ and $S \cdot S^T$. After the singular value decomposition of the projection normal matrix is known, evaluation of equation 49 uses $O(\Theta^2 K^2)$ operations.

For a rotationally invariant system, the evaluation of the projection normal matrix requires only $\Theta(\Theta K^2)$ integral evaluations and its singular value decomposition requires $\Omega(\Theta K^3)$ operations using the block circulant singular value decomposition algorithm described in section 3.2. The rotationally invariant orthonormal least squares estimator is computationally tractable on current computer systems.

[2] Let $n, n_0 \in \mathbf{N}$ and $\epsilon \in \mathbf{R}, \epsilon > 0$. Also, $f, g : \mathbf{N} \rightarrow \mathbf{R}$. Then, define [31]

1. Upper bound

$$O(f(n)) \equiv \{g(n) : g(n) \leq \epsilon f(n) \; \forall \, n > n_0\}$$

2. Lower bound

$$\Omega(f(n)) \equiv \{g(n) : g(n) \geq \epsilon f(n) \; \forall \, n > n_0\}$$

3. Combined bound

$$\Theta(f(n)) \equiv O(f(n)) \cap \Omega(f(n))$$

4. Asymptotic

$$f(n) \sim g(n) \Leftrightarrow \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 1$$

[3] Using the symbol Θ for the number of projection angles measured and the combined bound function $\Theta(\cdot)$ is somewhat confusing but parenthesis distinguish between the two uses.

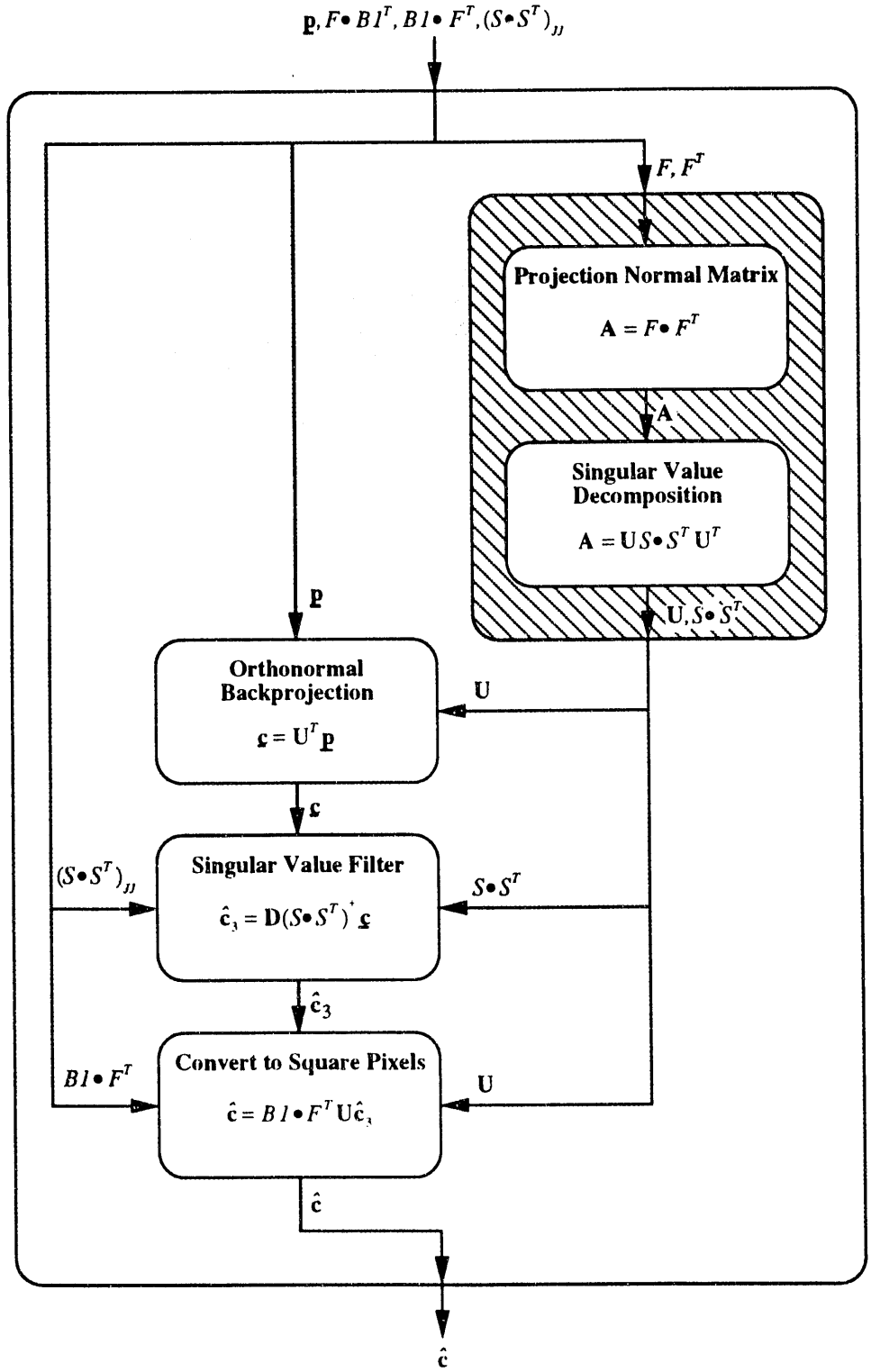

Figure 12: Block diagram of orthonormal natural pixel least squares estimation algorithm.

3.2 Block Circulant Singular Value Decomposition (BCSVD)

When the elements of the projection normal matrix are a function of only the difference $\Delta\theta = \theta' - \theta$ modulo Θ as shown in equation 50, the system is *rotationally invariant.* If it is not a function of the difference between k and k' modulo K, then the system is *radially variant.* The system is *spatially variant* if it is rotationally or radially variant.

$$
\begin{aligned}
A_{\theta' k' \theta k} &= A_{[(\theta-\theta')\bmod\Theta]k'0k} && (50)\\
&= A_{[\Delta\theta \bmod\Theta]k'0k} && (51)\\
&= \int_{\mathbf{R}} dy \int_{\mathbf{R}} dx\, f_{[\Delta\theta\bmod\Theta]k}(x,y)\, f_{0k}(x,y) && (52)
\end{aligned}
$$

When the system is rotationally invariant, the projection normal matrix can be written in block circulant form [32]. The block circulant structure is

$$
\boldsymbol{A} = \begin{bmatrix}
A_0 & A_1 & A_2 & .. & A_{\Theta-2} & A_{\Theta-1} \\
A_{\Theta-1} & A_0 & A_1 & .. & A_{\Theta-3} & A_{\Theta-2} \\
A_{\Theta-2} & A_{\Theta-1} & A_0 & .. & A_{\Theta-4} & A_{\Theta-3} \\
\vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\
A_2 & A_3 & A_4 & .. & A_0 & A_1 \\
A_1 & A_2 & A_3 & .. & A_{\Theta-1} & A_0
\end{bmatrix}. \qquad (53)
$$

There are $\Theta \times \Theta$ blocks each of size $K \times K$.

For the simple system of figure 4 the projection normal matrix can be written in block circulant form by reversing the direction of the projection axis for projection angle $\theta = 1$. With this change,

$$
\boldsymbol{A} = \frac{\pi}{6} \begin{bmatrix}
3 & 0 & 1 & 2 & 1 & 2 \\
0 & 3 & 2 & 1 & 2 & 1 \\
1 & 2 & 3 & 0 & 1 & 2 \\
2 & 1 & 0 & 3 & 2 & 1 \\
1 & 2 & 1 & 2 & 3 & 0 \\
2 & 1 & 2 & 1 & 0 & 3
\end{bmatrix}. \qquad (54)
$$

There are three by three blocks each of size two by two.

By using the rotational invariance properties of the projection normal matrix, computationally fast and efficient algorithms have been implemented for the reconstruction procedures described in section 1. As an example for the rest of this section, the case where $\Theta = K = 64$ shall be used because it is representative of some typical tomographic inverse problems.

A $\Theta(K^2\Theta \log \Theta)$ fast Fourier transform (FFT) technique [33] [34] and an $\Omega(\Theta K^3)$ singular value decomposition (SVD) algorithm are used to compute the factorization [35]

$$\begin{aligned} \boldsymbol{A} &= (\mathcal{F}_\Theta \otimes \boldsymbol{I}_K)^\dagger \; \boldsymbol{D} \; (\mathcal{F}_\Theta \otimes \boldsymbol{I}_K) && (55)\\ &= (\mathcal{F}_\Theta \otimes \boldsymbol{I}_K)^\dagger \; \boldsymbol{U_D \Lambda_D U_D^\dagger} \; (\mathcal{F}_\Theta \otimes \boldsymbol{I}_K) && (56)\\ &= \boldsymbol{U \Lambda_D U^T} && (57)\\ &= \boldsymbol{US} \cdot \boldsymbol{S^T U^T} && (58) \end{aligned}$$

where $\mathcal{F}_\Theta$ is a normalized $\Theta \times \Theta$ discrete Fourier operator matrix and $\boldsymbol{I}_K$ is $K \times K$ identity matrix.[4] The operator † is conjugate transpose and $\otimes$ is the outer product operation.

Each of the K^2 discrete Fourier transforms of equation 56 can be computed independently; i.e., each sum does not need the result or input of another sum [36] [37]. The SVD of the blocks of $\boldsymbol{D}$ also do not have input/output dependencies with other blocks and can be computed without explicit synchronization. Therefore, parallel processing implementations of the block circulant singular value decomposition algorithm are possible.

Two parallel versions [38] of the BCSVD algorithm were implemented and tested on a Cray-2 supercomputer using macrotasks [35]. The $\Theta(\Theta \log \Theta)$ grain size of FFT tasks is extremely small. For the example, it takes about 0.45 ms [39] [40]. This is comparable to the 0.31 ms necessary to synchronize with a server process and is much smaller than the 2.63 ms necessary to create a new process. It is thus advantageous to increase the grain size of FFT tasks by computing K FFTs per task. The resulting granularity of $\Theta(K\Theta \log \Theta)$ is about 29 ms. The task granularity of an SVD process is $\Omega(K^3)$ which is 428 ms for the example problem.

A prescheduled algorithm was implemented by creating one process for each of the K FFT tasks and another process for each of the Θ SVD tasks. The parent task starts n processes with either an FFT or an SVD task. All of the n processes run to completion before another n processes are started. This method is very easy to implement because all synchronization is implicit in the fork and join like paradigm [41].

To overcome the process creation overhead, a self-scheduling algorithm was constructed [38]. This method is more complex than the prescheduled algorithm but has a smaller time overhead. It requires explicit synchronization between server processes and a task manager. n server processes are created and each waits for a start signal after initial setup of local state information. After receiving the start signal from the task manager, a server checks what part of the matrix it is to work on next. When finished the server sends a ready signal to the hibernating manager. The manager then reassigns each of the server processes until the task queue is empty.

Figure 13 shows the computation time for different sizes of input matrices. The speedup of the algorithm, shown in figure 14, increases as the size of Θ and K are increased. The prescheduled algorithm is faster for very small matrix sizes because the self-scheduled algorithm server processes have a larger startup overhead than a process started by the prescheduled algorithm. The self-scheduled algorithm is faster for medium sized problems that have small grain sizes but the prescheduled algorithm again approaches the speedup of self-scheduling as the problem size increases.

[4]The matrix $\boldsymbol{A}$ of equation 53 is less general than the algorithm will accept since it is symmetric with square blocks and real elements instead of having rectangular blocks and complex elements.

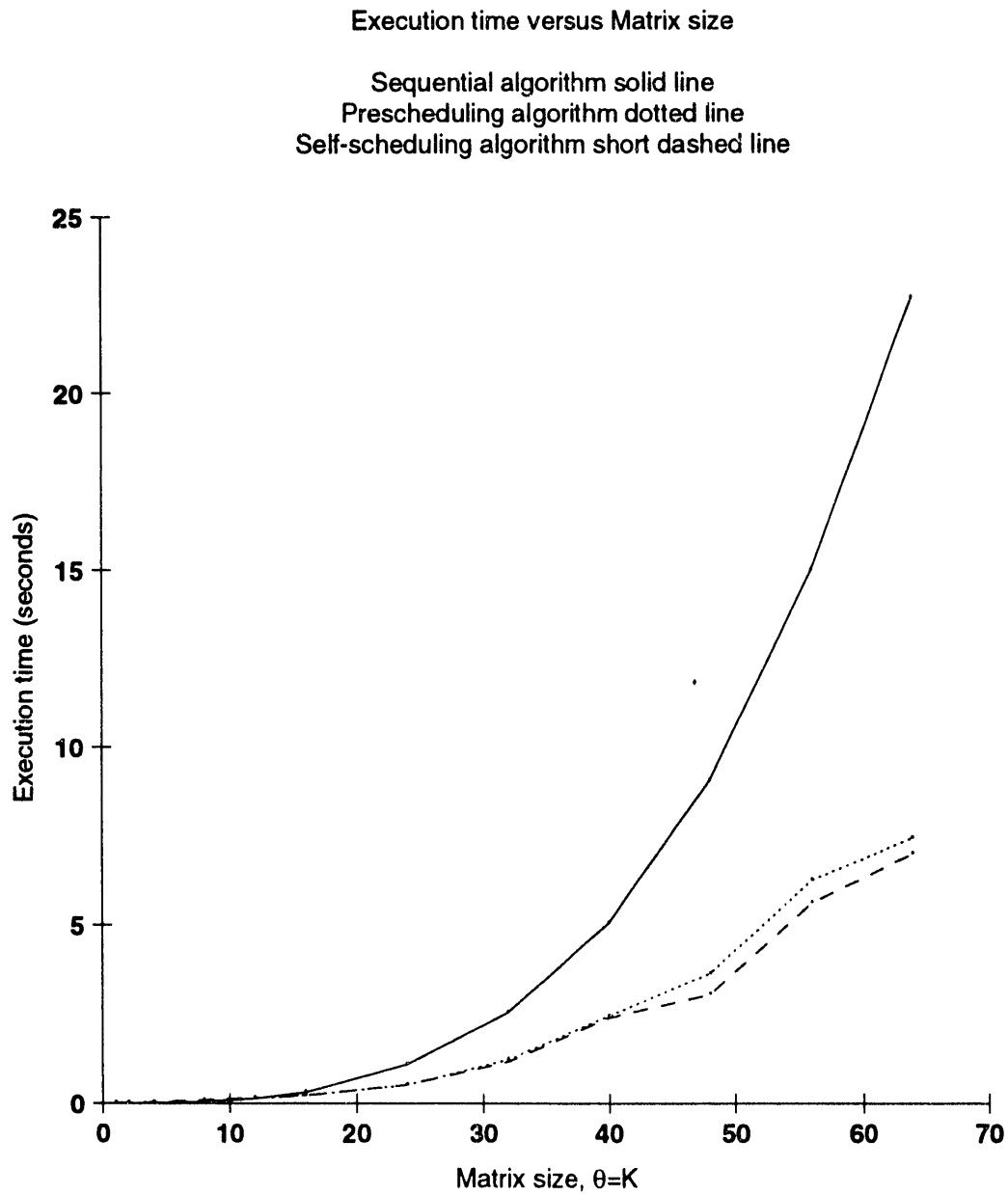

Figure 13: Computation time versus problem size with four tasks and four processors available to service tasks.

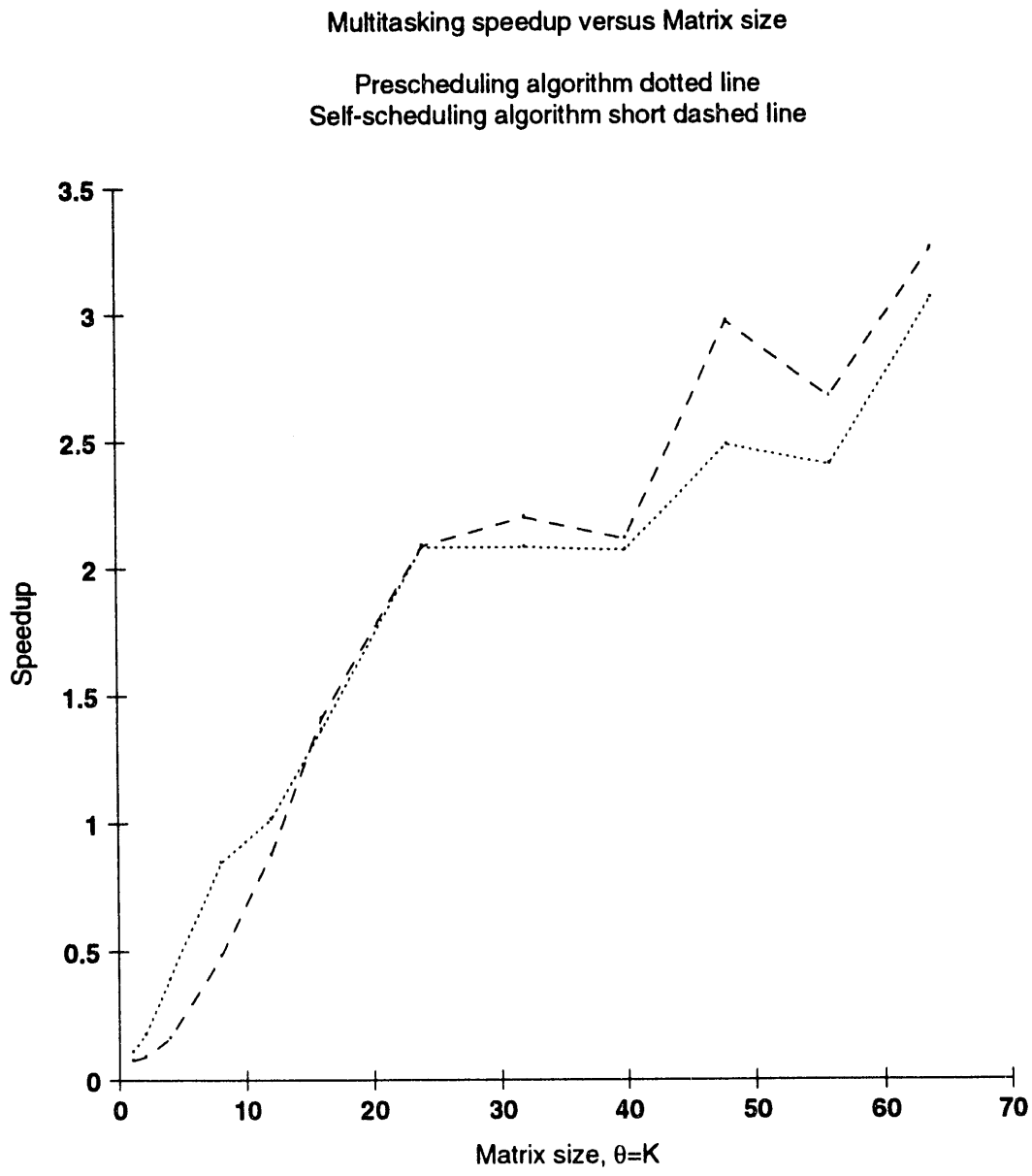

Figure 14: Speedup versus problem size with four tasks and four processors available to service tasks.

The efficiency, shown in figure 15, does not approach unity as quickly as expected. This might be attributed to the timesharing scheduling algorithm used by the CTSS operating system and not to synchronization overhead because the overhead, shown in table 4, is less than 1.0% for Θ and K larger than 64 [42] [43]. It was not possible to verify this conjecture by using the machine without other users present.

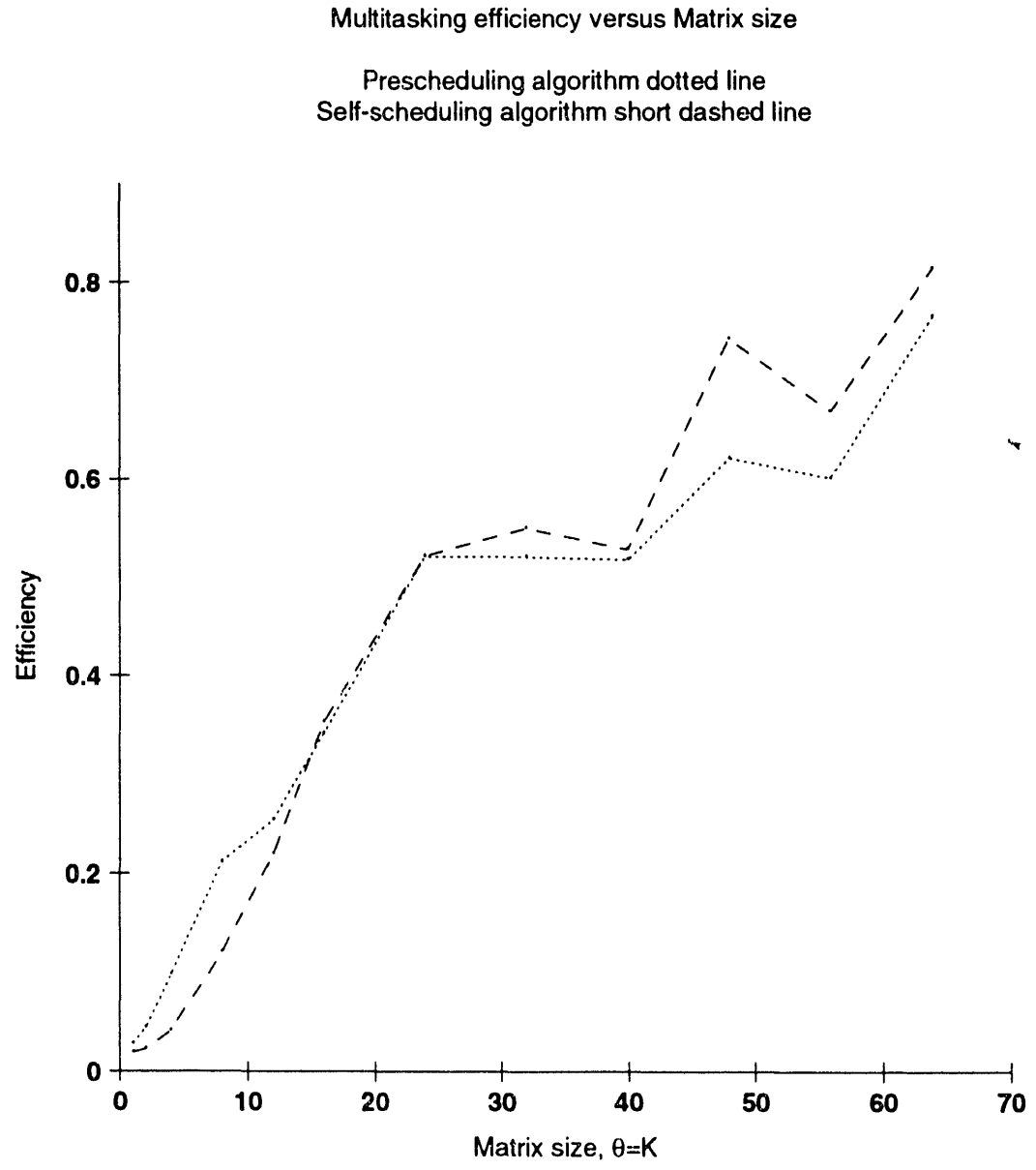

Figure 15: Efficiency versus problem size with four tasks and four processors available to service tasks.

The process creation time was found to be 2.63 ms. Task synchronization in the self-scheduling algorithm was 0.31 ms. A typical procedure call was measured to take 4.7 μs. Self-scheduling has less time overhead than prescheduling but is still 66 times more expensive than a procedure invocation.

Data memory usage and overhead is shown in table 5. Very little memory is necessary

$\Theta = K$	prescheduled		self-scheduled	
	overhead (ms)	% overhead	overhead (ms)	% overhead
4	25.7		29.9	
8	46.8	42.77	32.3	5.09
16	80.2	10.89	37.3	1.29
32	155.7	2.66	47.2	0.32
64	306.8	0.61	67.1	0.07

Table 4: Synchronization overhead versus problem size.

for the synchronization of tasks. Each of the processes needs some local working storage for computing FFTs and SVDs. Code memory usage and overhead is shown in table 6. The code space sharing was small due to a problem in the Fortran compiler that made code replication necessary.

$\Theta = K$	sequential	prescheduled		self-scheduled	
	usage (kB)	usage (kB)	% overhead	usage (kB)	% overhead
8	112	409	265.2	475	324.1
16	240	533	130.4	604	151.7
32	1648	1946	18.1	1948	18.2
64	12400	12698	2.4	12888	3.9

Table 5: Data memory usage and overhead versus problem size with four tasks.

n	sequential	prescheduled		self-scheduled	
	usage (kB)	usage (kB)	% overhead	usage (kB)	% overhead
1	404	450	11.4	447	10.6
2	404	489	21.0	492	21.8
3	404	530	31.1	537	32.9
4	404	570	41.1	582	44.1

Table 6: Code memory usage and overhead versus number of active tasks.

Dynamic memory allocation costs are basically independent of the block size being allocated for small blocks. The cost depends almost entirely on the number of blocks being allocated. Each block takes approximately 0.68 ms to allocate. The server processes of the self-scheduling algorithm avoid this overhead by reusing their local storage during each activation. The prescheduling algorithm originally allocated local storage blocks within each child process. This was deemed to be unsatisfactory and another parameter with working storage was passed to each child to avoid the overhead of dynamic memory allocation.

The BCSVD algorithm provides orders of magnitude speedup by utilizing the circulant structure of matrices. A further speedup was obtained using macrotasking. This does not reduce central processing unit charges because time on all processors is billed to the job [43]. However, a substantial savings in memory charges is achieved because the program memory residency time is reduced by the multiprocessor speedup [44] [45]. For typical problems Θ and K are approximately 256. This requires approximately 800 megabytes of memory which can be quite costly to use.

Self-scheduling is useful when the task granularity is small. As the task granularity increases, prescheduling overhead becomes less important. Prescheduling is much easier to implement and debug. There are no explicit synchronizations to consider since the operating system handles the process allocation and scheduling. The parent only has to wait for the operating system to signal that the child has finished. Self-scheduling needs explicit synchronization with the server tasks and is therefore more difficult to implement and debug.

The Fortran compiler does not allocate local variables on the stack properly. It puts some local variables into static storage. Thus, code sharing is not possible for the Fortran subroutines. Each process must have a separate copy of the code and local data space. This was done by creating copies of the subroutines and giving each copy a unique name space by appending the process number to the name of the subroutine and all of its descendants.

4 Conclusions

The inverse problem requires the solution of large systems of linear and non-linear equations. For example, the Donner 600-Crystal Positron Tomograph takes 120,200 projection measurements and the resulting linear system is 120,200 $\times$ 120,200. The computational complexity of configuration space methods based on orthonormal natural pixels has led to the use of a distributed computing environment in which workstations are used to analyze results from our identification, estimation, and optimization algorithms running on supercomputers in a multitasking environment with priority scheduling. Several of the algorithms have been implemented using large grain parallel processing and also remote procedure calls.

The block circulant singular value decomposition (BCSVD) algorithm uses discrete Fourier transforms to rotate the blocks of a block circulant matrix into block diagonal form. Each block on the diagonal is then factored using a general singular value decomposition (SVD) algorithm. The BCSVD algorithm provides orders of magnitude speedup over general SVD algorithms. For a $64^2 \times 64^2$ block circulant matrix, computation time decreased from 12 hours to 23 seconds on a Cray-2 (approximately 1 hour on a SPARCstation II) . Because the BCSVD algorithm is easy to partition, a further speedup can be achieved using parallel processing. The orthogonality properties of multidimensional fast Fourier transforms (FFT) allows the FFT portion of the algorithm to partition into macrotasks. The SVD of the blocks of the block diagonal matrix can be computed independently and a macrotask can be assigned to each SVD. A multiprocessor speedup of 3.06 was achieved for prescheduling and for self-scheduling a multiprocessor speedup of 3.25 was observed using four processors on a Cray-2. Relative time overhead was 0.5% for the prescheduled algorithm and 0.07% for the self-scheduled algorithm. Relative memory overhead was 4% for both cases. Self-scheduling is useful when the task granularity is small. As the task granularity increases, prescheduling

ing overhead becomes less important. The prescheduled algorithm is satisfactory for most emission tomography problems because all the dimensions of the matrices are greater than 64 and the task granularity will therefore be large when compared to the synchronization overhead.

Multitasking the block circulant singular value decomposition algorithm decreases overall computation costs by reducing the time large sections of memory are in use. Little or no gain comes from reduced central processing unit charges since processing time on all processors is charged to a job.

The data acquisition model presented in this work provides a mathematical framework to incorporate detailed knowledge about the response function of a tomography system and the statistical properties of the signals acquired using that system. Using this mathematical forward problem model, it is easy to represent systems of varying complexity; e.g., simple spatially invariant systems, systems with spatially variant response that have symmetries, and the most general linear case, a spatially variant system without symmetry. The novel singular value decomposition of the projection formation operator used in the data acquisition model is a powerful mathematical description of a tomography system and is fundamental to the estimation (inverse problem) methods presented here. While the results presented in the examples are not inconsequential, these techniques should find the most utility in modeling the spatial sampling of each unique tomograph. Extensions of the model to include sampling in three spatial dimensions as well as time should be straightforward; but, in practice, higher dimensional applications may be limited by computational tractability without using special computing technology.

Because the data acquisition process is represented as a linear map from a continuous domain object space to a discrete domain observation space, it is a more physically realistic model of many systems than approximations using continuous-continuous maps or discrete-discrete maps. Thus, the validity of many results that were obtained using these approximations, e.g., angular and lateral sampling density in emission tomography, may need to be reexamined using the new, more robust techniques presented in this work. While the verification of old results is worthwhile, it is the unanswered questions such as the efficacy of iterative algorithms and stopping rules, the formulation of Bayesian estimators that use basis functions in the null space of the projection formation operator, the representation of object functions that are convex cones, and the efficiency of algorithm implementations that provide challenging new research opportunities for the application of the mathematical methods presented here to characterize and solve inverse problems in tomography.

The computational complexity of reconstruction algorithms can be reduced by using orthonormal natural pixels because this basis preserves symmetries in the data acquisition process. The present state of the art in computing hardware can be expected to accomplish a properly composed orthonormal natural pixel least squares reconstruction of data taken with a modern rotationally invariant tomography system, such as the Donner 600-Crystal Positron Tomograph, in 13 minutes per tomograph slice. While substantial performance enhancements are often realized using improved hardware, specialized thought in algorithm development can lead to more significant performance increases allowing the solution of previously intractable problems like those found in tomography.

References

[1] Andrews HC and BC Hunt. *Digital Image Restoration.* Prentice-Hall, Englewood Cliffs, NJ, 1977.

[2] Baker JR and TF Budinger. Advanced models for medical imaging. In *High Speed Computing: Scientific Applications and Algorithm Design*, pages 221–226, Urbana, IL, 1987.

[3] Hanson KM. Bayesian and related methods in image reconstruction from incomplete data. In Stark H, editor, *Image Recovery: Theory and Application*, pages 79–123. Academic Press, Orlando, 1987.

[4] Baker JR. Spatially variant tomographic imaging: Estimation, identification, and optimization. Ph.D. Thesis LBL-31561, University of California, Berkeley, 1991.

[5] Luenberger DG. Optimization by vector space methods. In Howard RA, editor, *Decision and Control.* John Wiley and Sons, Inc., New York, 1969.

[6] Bellini S, M Piacentini, C Cafforio, and F Rocca. Compensation of tissue absorption in emission tomography. *IEEE Trans ASSP*, **ASSP-27**(3):213–218, 1979.

[7] Jaszczak RJ and RE Coleman. Single photon emission computed tomography (SPECT) principles and instrumentation. *Invest Radiol*, **20**:897–910, 1985.

[8] Huesman RH, EM Salmeron, and JR Baker. Compensation for crystal penetration in high resolution positron tomography. *IEEE Trans Nucl Sci*, **NS-36**:1100–1107, 1989.

[9] Radon J. Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiter. *Berichte Sächsische Akademie der Wissenschaften*, **69**:262–267, 1917.

[10] Cormack AM. Representation of a function by its line integrals, with some radiological applications. *J Appl Phys*, **34**(9):2722–2727, 1963.

[11] Cormack AM. Representation of a function by its line integrals, with some radiological applications II. *J Appl Phys*, **35**(10):2908–2913, 1964.

[12] Deans SR. *The Radon Transform and some of its Applications.* John Wiley & Sons, New York, 1983.

[13] Goitein M. Three-dimensional density reconstruction from a series of two-dimensional projections. *Nucl Instrum Meth*, **101**:509–518, 1972.

[14] Hsieh RC and WG Wee. On methods of three-dimensional reconstruction from a set of radioisotope scintigrams. *IEEE Trans Systems, Man, Cybernetics*, **SMC-6**(12):854–862. 1976.

[15] Budinger TF, GT Gullberg, and RH Huesman. Emission computed tomography. In Herman GT, editor, *Image Reconstruction from Projections*, volume 32, pages 147–246. Springer-Verlag, New York, NY, 1979.

[16] Wood SL. A system theoretic approach to image reconstruction, May 1978. Ph.D. Thesis.

[17] Levitan E and GT Herman. A maximum *a posteriori* probability expectation algorithm for image reconstruction in emission tomography. *IEEE Trans Med Imaging*, **MI-6**:185–192, 1987.

[18] Leahy RM and CE Goutis. An optimal technique for constraint-based image restoration and reconstruction. *IEEE Trans Acoust, Speech, Signal Processing*, **ASSP-34**(6):1629–1642, 1986.

[19] Leahy R, TJ Hebert, and R Lee. Applications of Markov random fields in medical imaging. *Prog Clin Biol Res*, **363**:1–14, 1991.

[20] Gordon R, R Bender, and GT Herman. Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography. *J Theor Biol*, **29**:471–481, 1970.

[21] Golub GH and C Reinsch. Singular value decomposition and least squares solutions. *Numer Math*, **14**:403–420, 1970.

[22] Lawson CL and RJ Hanson. *Solving Least Squares Problems.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[23] Golub GH and CF Van Loan. *Matrix Computations*, volume 3. Johns Hopkins University Press, Baltimore, MD, 1983.

[24] Buonocore MH, WR Brody, and A Macovski. A natural pixel decomposition for two dimensional image reconstruction. *IEEE Trans Biomed Eng*, **BME-28**(2):69–78, 1981.

[25] Marr RB. On the reconstruction of a function on a circular domain from a sampling of its line integrals. *J Math Anal Appl*, **45**:357–374, 1974.

[26] Moore EH. *Bull Amer Math Soc*, **26**:394–395, 1920.

[27] Penrose R. A generalized inverse for matrices. *Cambridge Philosophical Soc*, **51**:406–413, 1955.

[28] Strang G. *Linear Algebra and Its Applications.* Academic Press, Orlando, FL, 1980.

[29] Floyd CE, RJ Jaszczak, and RE Coleman. Image resampling on a cylindrical sector grid. *IEEE Trans Med Imaging*, **MI-5**(3):128–131, 1986.

[30] Trussell HJ, H Orun-Ozturk, and MR Civanlar. Errors in reprojection methods in computerized tomography. *IEEE Trans Med Imaging*, **MI-6**:220–227, 1987.

[31] Knuth DE. *The Art of Computer Programming*, volume 1. Addison Wesley, Reading, MA, 1981.

[32] Davis PJ. *Circulant Matrices.* John Wiley and Sons, New York, NY, 1979.

[33] Brigham EO. *The Fast Fourier Transform.* Prentice-Hall, Englewood Cliffs, NJ, 1974.

[34] Bracewell RN. Fourier techniques in two dimensions. In Price JR, editor, *Fourier Techniques and Applications*, volume 3, part 1, pages 45–71. Plenum Press, New York, 1985.

[35] Baker JR. Macrotasking the singular value decomposition of block circulant matrices on the Cray-2. In *Proceedings of Supercomputing '89*, pages 243–247, Reno, NV, 1989.

[36] Zenios SA and Y Censor. Parallel computing with block-iterative image reconstruction algorithms. Medical Image Processing Group Technical Report MIPG136, Department of Radiology University of Pennsylvania, October 1988.

[37] Herman GT, D Odhner, KD Toennies, and SA Zenios. A parallelized algorithm for image reconstruction from noisy projections. Medical Imaging Process Group Technical Report MIPG155, Department of Radiology University of Pennsylvania, September 1989.

[38] Quinn MJ. *Designing Efficient Algorithms for Parallel Computers.* McGraw-Hill, New York, NY, 1987.

[39] Buneman O. Vector FFT for the Cray-2. *NMFECC (National Magnetic Fusion Energy Computing Center) Buffer*, **10**(11):10–11, 1986.

[40] Despain AM. Very fast Fourier transform algorithms for hardware implementation. *IEEE Transactions on Computers*, **C-28**(5):333–341, 1979.

[41] Mundie DA and DA Fisher. Parallel processing in Ada. *Computer*, **19**(8):20–25, 1986.

[42] Mirin A. Parallelization of a 3-d MHD code, part I: Methodology and results. *NMFECC (National Magnetic Fusion Energy Computing Center) Buffer.* **11**(7):14–16, 1987.

[43] Mirin A. Parallelization of a 3-d MHD code, part II: Analysis of multiprocessing efficiency on the Cray-2. *NMFECC (National Magnetic Fusion Energy Computing Center) Buffer*, **11**(8):11–13, 1987.

[44] Patton PC. Multiprocessors: Architectures and applications. *Computer*, **18**(6):29–40, 1985.

[45] Gelernter D. Domesticating parallelism. *Computer*, **19**(8):12–16, 1986.

END

DATE FILMED

2/23/93