CT Medical Imaging Reconstruction Using Direct Algebraic Methods with Few Projections

Mónica Chillarón[1]([✉]) [iD], Vicente Vidal[1] [iD], Gumersindo Verdú[2] [iD], and Josep Arnal[3] [iD]

[1] Departamento de Sistemas Informáticos y Computación (DSIC), Universitat Politècnica de València, Valencia, Spain
mnichipr@inf.upv.es, vvidal@dsic.upv.es
[2] Instituto de Seguridad Industrial, Radiofísica y Medioambiental (ISIRYM), Universitat Politècnica de València, Valencia, Spain
gverdu@iqn.upv.es
[3] Departamento de Ciencia de la Computación e Inteligencia Artificial (DCCIA), Universidad de Alicante, Alicante, Spain
arnal@ua.es

Abstract. In the field of CT medical image reconstruction, there are two approaches you can take to reconstruct the images: the analytical methods, or the algebraic methods, which can be divided into iterative or direct.

Although analytical methods are the most used for their low computational cost and good reconstruction quality, they do not allow reducing the number of views and thus the radiation absorbed by the patient.

In this paper, we present two direct algebraic approaches for CT reconstruction: performing the Sparse QR (SPQR) factorization of the system matrix or carrying out a singular values decomposition (SVD). We compare the results obtained in terms of image quality and computational time cost and analyze the memory requirements for each case.

Keywords: CT · Medical imaging · Reconstruction
Matrix factorization · QR · SVD · Few projections

1 Introduction and Background

In medical imaging, CT (computerized tomography) [18,24] is one of the most significant tests to perform a diagnosis. Thus, it is imperative to develop reconstruction algorithms that provide high-quality images as well as high computational time efficiency. Having fast algorithms is essential to have short-term results of a CT scan available to medical professionals.

The reconstruction methods can be divided into two types, depending on their nature. On the one hand, we find the analytical algorithms. They are based on the application of the Fourier transform on the data obtained from the

projection of x-rays on an object, called sinogram. On the other hand, we have the algebraic methods, whether direct or iterative, which make a mathematical approach to the reconstruction problem.

In clinical practice, the most widespread methods are the analytical ones. This is due to the reduced computational time cost involved, which can be vital in emergency diagnoses. However, algebraic methods allow reducing the number of projections, and therefore the radiation to which we expose the patient.

Since hardware elements evolve at high speed and also their cost decreases constantly, using algebraic approaches to solve these problems has become possible. Although they involve a high computational cost, nowadays it is less significant thanks to parallel computing (multiple CPUs, GPU computing, clusters, etc.) [12]. If we add the enhancement of main memory in new equipments, it is now feasible to implement these methods for large size problems as is our case.

The aim of this work is to achieve the resolution of the CT image reconstruction problem by means of two direct algebraic factorization methods. The first one, called multifrontal sparse QR (SPQR), and implemented in the library 'SuiteSparseQR, a multifrontal multithreaded sparse QR factorization package' [7], allows solving the equation of reconstruction problem more directly than the iterative methods we studied previously (such as ART [1], SART [2] or LSQR [22]). This is achieved by calculating the QR factorization of the sparse system matrix A to simulate its pseudo-inverse.

The second method is the singular values decomposition (SVD) of the system matrix A, which is carried out through the parallel implementation of the method included in the SLEPc [14] to simulate, as in the previous case, the pseudo-inverse of the matrix.

Since these methods are not widespread in clinical practice, the validity of both approaches will be tested. We will check the quality of the reconstructed image by working with a smaller number of projections in order to reduce the radiation induced to the patient, and analyze the computation resources necessary to perform the decomposition of the matrix A. This point will be vital for the use of these algebraic methods, since the matrices are large and can suppose a high demand for RAM.

2   Materials and Methods

2.1   CT Image Reconstruction Problem

In a CT reconstruction problem, using an algebraic approach we need to solve the equations system defined by the Eq. (1), where A (2) represents the system matrix, a sparse matrix that measures the weight of the influence that each ray has on the reconstructed image [5,16]. The size of the matrix A is M × N, where M is the number of traced rays, and N is the size in pixels of the image to be reconstructed. The vector g (3) represents the sinogram or projections vector and u (4) the solution image. Both the system matrix and the sinogram vector have been calculated with Joseph's Forward Projection method [17].

$$A * u = g \tag{1}$$

$$A = a_{i,j} \in \mathbb{R}^{MxN} \tag{2}$$

$$g = [g_1, g_2, \ldots, g_M]^T \in \mathbb{R}^M \tag{3}$$

$$u = [u_1, u_2, \ldots, u_N]^T \in \mathbb{R}^N \tag{4}$$

The projections have been generated for a phantom corresponding to the mathematical representation of the human head developed by the Forbild Phantom Group [10], using the medical image program CONRAD [20] to generate the phantom reference images. This is defined by simple geometric objects that represent head elements with different densities such as bone, tissue, gray matter, etc.

When reconstructing a CT image for a given physical configuration of the scanner, the associated matrix A is always the same. What changes is the sinogram (g), that represents the studied object. If it were possible to calculate a pseudo-inverse A^+ of the matrix A [13,19], the image could be obtained more directly without solving a large equations system. This means the computational cost would be reduced to a matrix-vector product. But this idea not viable since the explicit calculation of the matrix A^+ is prohibitive for a high resolution and many views in the reconstruction. The reason why is that A^+ can be dense and contaminated due to rounding errors.

Another possibility is to calculate implicitly an approximation of the range and use it to simulate the pseudo-inverse. The two methods that we have considered to solve this approach are explained in the following subsections

2.2 Projections Reduction

In the acquisition of the data, the important factors that influence the reconstruction of the image are two, the number of samples per projection (detectors) and the number of projections. A reduced number of any of these two variables will cause the formation of artifacts in the reconstructed image when using analytical methods: rings by the Gibbs phenomenon, streaks and lines and Moiré pattern, amongst others [3].

To prevent such problems, we should take into account the Nyquist theorem [21], which says the sampling rate must be greater than twice the bandwidth of the sampled signal. If we undersample we get the aforementioned artifacts. The classical analytical procedure of image reconstruction (filtered back projection FBP) is a fast process but it needs a complete set of projections to obtain high-quality images. The exact number depends on the physical characteristics of the scanner, but typically the minimum number of projections to take is 360.

If the aim is to reduce the dose absorbed by the patient, we should take the fewer number of projections possible, so we should use different methods that can work with less projections, such as iterative or direct methods. In our previous works [6,8,9,23] it has been shown that it is possible to use few views, between 30 and 90, while using iterative methods to reconstruct high-quality images. However, the computational cost is high for these algorithms.

In order to take an approach that is not iterative, we propose the SPQR and SVD methods. In this way, the computational cost of the reconstruction could be reduced, so we could reconstruct images faster in real time. We will check if its application to the CT image reconstruction allows reducing the number of views in equal measure than the methods that we used previously.

2.3 Sparse Pivoting QR Factorization

The first new direct method used to solve the equations system is the Multi-frontal Sparse QR (SPQR) [7]. It is a method that performs a QR [13] factorization of a large sparse matrix in a sequence of dense frontal matrices. In this way, we can form the pseudo-inverse of the system matrix.

The software used to perform this factorization is implemented in the SuiteS-parseQR library, which uses BLAS and LAPACK, and Intel Threading Building Block to exploit parallelism. This allows processing heavy matrices, which is key in the reconstruction of CT images since A is large for high resolutions.

The QR factorization of the matrix A comprises its decomposition as a product of two matrices (5), where Q is orthogonal and R is upper triangular.

$$A = QR \tag{5}$$

$$AP = QR \tag{6}$$

However, when A is considerably large and sparse, this decomposition can be prohibitive since Q could not be sparse. Here, the decomposition with pivoting (6) must be used. The resolution would be:

– If $m \geq n$ we have to solve (7). That means a matrix-vector product of Q^T by the vector g, the resolution of an upper triangular system with the matrix R and a rearrangement with permutation matrix P.
– If $m < n$ we have to solve (8), that is reordering the vector g, then solve a triangular system and finally make the product of the resulting vector by the matrix Q.

$$u = P * (R^{-1}(Q^T * g)) \tag{7}$$
$$u = Q * (R^{-T}(P^T * g)) \tag{8}$$

2.4 Q-Less Sparse QR Factorization

If we want to spare memory, we can compute the QR decomposition without saving the Q explicitly. In this way we would have to transform the equations to obtain the solution vector u, which would be as follows:

– If $m \geq n$ we have to solve the system $Q * R * g = u$. By multiplying both sides of equation by A^T, we obtain: $(Q * R)^T * Q * R * g = A^T * u$. Since Q is orthogonal, performing the matrix operations to solve for u, we reach the solution Eq. (9).

$$u = R^{-1} * R^{-T}(A^T * g) \tag{9}$$

– If $m < n$ we would work with the QR decomposition of A^T. By a similar reasoning to the previous case, the solution of the system would correspond to calculate (10), which requires the same operations but in a different order.

$$u = A^T * R^{-1} * (R^{-T} * g) \tag{10}$$

Note that the Q matrix is not used for the calculation of u and therefore this process requires fewer memory resources.

2.5   Singular Values Decomposition

The singular values decomposition performs the k-order factorization (11) [4,11,19]. In this factorization, $U \in \mathbb{R}^{Mxk}$ and $V \in \mathbb{R}^{kxN}$ are matrices whose columns are orthogonal and are called left-hand and right-hand singular vectors, where k is the number of singular values computed. $\Sigma \in \mathbb{R}^{kxk}$ is a diagonal matrix that has the singular values in decreasing order. The software we use to apply this algorithm is the SVD solver included in the eigenvalues calculation parallel library SLEPc. Applying this factorization, we can transform the equations system into the product (13), taking the pseudo-inverse as (12)

$$A = U\Sigma V^T \tag{11}$$

$$A^+ = V\Sigma^{-1}U^T \tag{12}$$

$$u = A^+ g \tag{13}$$

Note that the storage cost of this decomposition is of order *(M * k) + (k * N)* elements. When k grows, the memory resources necessary to carry out the decomposition increase. Once the decomposition of the system matrix has been calculated, resolving the problem consists on the product of two dense matrices by a vector,thus the computational cost is of order *((M + N) * k)* flops.

Since we can do this decomposition 'offline', it can be calculated and stored and it can be ready when a CT image has to be reconstructed. Therefore, we can replace an iterative method with a much simpler and less computationally expensive matrix multiplication problem. In our application, we are interested in the case where $k = N$, in which case we have enough information to reconstruct the image with great precision.

3   Results and Discussion

In order to check the validity of both methods, we used a cluster belonging to the Universitat Politécnica de Valéncia to perform the factorizations. The cluster consists of 4 RX500S7 servers with four Intel Xeon E5-4620 8 core processors (32 cores per node) and 256 GB DDR3 RAM (8 GB/core ratio) per node.

3.1 Memory Requirements

For the SPQR factorization, we have used the code developed for Matlab. We reserve a single node in the cluster. In this node which we can consume a maximum of 250 GB of main memory. With these resources, it has been possible to compute the QR decomposition corresponding to the system matrix for an image resolution from 32×32 up to 256×256 with 30 views. Besides, using the Q-less QR factorization we have been able to use up to 90 views with the 256×256 resolution, which is important as we will explain in Sect. 3.3. For the analogous case of resolution 512×512, considering 30 views, the matrix has not been factorized due to lack of RAM.

Regarding the SVD decomposition, all cases of 30 views have been computed, except for the 512×512 resolution case. For this process, we have reserved up to 32 processors with 15 GB of RAM per processor, which means up to 240 GB. Despite making use of distributed memory, we can not reserve over one server node at a time, similar to the sequential case. In addition, the SVD process requires more memory than the SPQR, so with the same characteristics, it has not been possible to reach the case of 90 views for the 256×256 resolution.

3.2 Computational Time Efficiency

Although the factorization time is not critical in this scenario, it is convenient to have an approximation of the computation time required for each matrix size. In addition, we want to make a comparison between the two proposed methods. Table 1 shows the results in seconds of computational time cost to carry out the decompositions.

Table 1. Factorization time

Resolution	Factorization method	
	SPQR	SVD
	Time (secs.)	
32×32 30 views	1.2	9
64×64 30 views	5.6	50
128×128 30 views	151	1900
256×256 30 views	3270	12000
256×256 60 views	12336	-
256×256 90 views	16400	-

As we can observe, the factorization time in both cases is satisfactory. Although it may seem that with the SVD decomposition it is too high, we must bear in mind that this calculation will be performed only once and 'offline', which makes it a non-critical calculation. As shown in the table, the computational time efficiency of the SPQR method is greater, which can be justified

because this calculation is performed in a single node, and therefore does not require distributed memory as does the SVD. Since SVD implements the communication through MPI messages, it generates an extra time associated with interprocess communications, which makes the temporary efficiency much worse.

3.3 Image Quality

Regarding the quality of the images obtained by performing a reconstruction with both methods, the results are reflected in Table 2. To measure the quality, we used the PSNR metric [15], which shows the noise level of a image compared to a reference image, and also the SSIM metric [15], which indicates the structural similarity between both images, based on the shape of the elements it contains.

Table 2. Reconstructed images quality

Resolution	SPQR		SVD	
	PSNR	SSIM	PSNR	SSIM
32×32 30 views	228	1	238	1
64×64 30 views	213	1	224	1
128×128 30 views	255	1	221	1
256×256 30 views	38	0.03	30	0.29
256×256 60 views	32.5	0.03	-	-
256×256 90 views	150	1	-	-

In both cases we can see that up to the resolution 128×128, the results of the reconstruction are very good, getting a PSNR greater than 200 in all cases (considering that a PSNR close to 100 can be considered a perfect reconstruction to the perception of the eye human). In addition, all cases obtain an SSIM equal to 1, which means that structurally the image is accurate. The result of the reconstruction by SVD with resolution 128×128 can be seen in Fig. 1b. If we compare it with the reference image for the same resolution (Fig. 1a), we observe that they are almost identical and the reconstruction does not have artifacts or noise. However, for the 256×256 resolution and 30 views, the result is of very poor quality for both methods, being the PSNR of both around 30. The reconstructed images, that we can see in Fig. 2 are very noisy and although we get to perceive elements of the phantom, it is not seen clearly enough.

The poor quality of this reconstruction is due to the fact that for this particular case (256×256 and 30 views), the sub-matrix extracted from the system matrix is rank deficient, as shown in Table 3. Therefore, the equations system is ill-conditioned, which leads to missing information necessary for the reconstruction. We observe that in the rest of the cases with smaller resolutions the sub-matrix retains the full rank, so we have no problem when reconstructing using these algebraic methods.
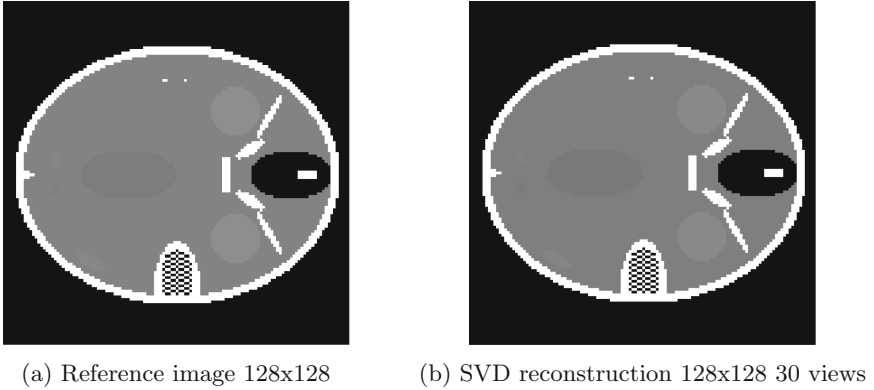
(a) Reference image 128x128   (b) SVD reconstruction 128x128 30 views

Fig. 1. Reference and reconstructed 128×128 images

(a) SPQR reconstruction   (b) SVD reconstruction
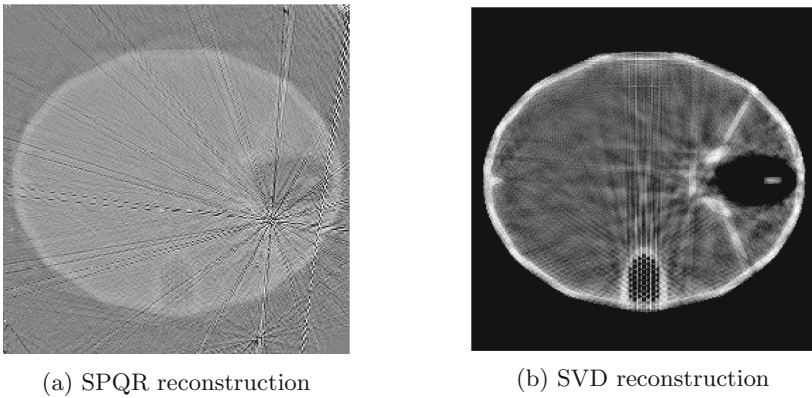
Fig. 2. Reconstructions 256×256 30 views

If we analyze the rank of the sub-matrices when we work with the largest resolutions, we can observe that we need to use more views in order to reach full rank. For 256×256 pixels reconstructions, we should use at least 90 views, which means we still reduce the number of views with respect to traditional methods. For 512×512 resolution, we need over 180 views, approximately 260, to keep the full rank on the sub-matrix. That means that even if we could still reduce a few views compared with other methods, the resulting reconstruction problem would have larger dimensions than we can compute.

We have verified that the rank of the matrix used has direct effect on the reconstructed image. In Fig. 3 we present the singular vector of a full-rank sub-matrix. As we can see, we have a few dominant values, approximately 1000, and then they start to decrease. When we reach the 8000th value, it looks like they stabilize and are close to 0. If we look to the detail window on the plot, we can observe that the last 1600 values, even if they are close to 0, vary between 0.03 and 0.01, which is still a significant number in this case. Therefore, this means we can not disregard any singular value.

Table 3. Rank study

| Resolution	Number of views				
	180	120	90	60	30
	N° of columns/rank				
32×32	1024/1024	1024/1024	1024/1024	1024/1024	1024/1024
64×64	4096/4096	4096/4096	4096/4096	4096/4096	4096/4096
128×128	16384/16384	16384/16384	16384/16384	16384/16384	16384/16384
256×256	65536/65536	65536/65536	65536/65536	65536/49380	65536/30093
512×512	262144/149551	262144/100842	262144/76000	262144/50963	262144/25608
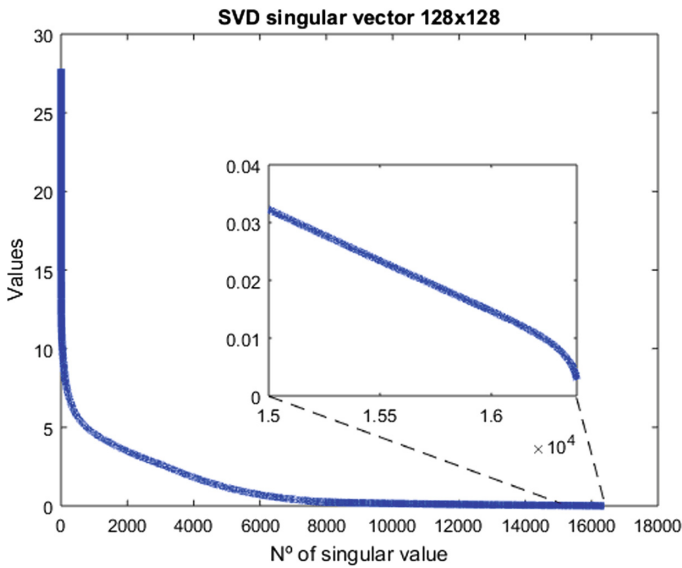
Fig. 3. Singular vector 128×128

In Fig. 4 we can observe how varying the number of singular values, we get very different reconstructions. As explained before, we do not reach optimal quality until we use the full rank. We observe the same effect with SPQR if we vary the number of views.

In Fig. 5 we see the difference between taking 30, 60, and 90 for the 256×256 resolution matrix. With 60 projections we increase the rank, so the image is slightly better than with 30, as can be observed in the edges of the phantom. But it is not until we reach 90 views that the image is of good quality.
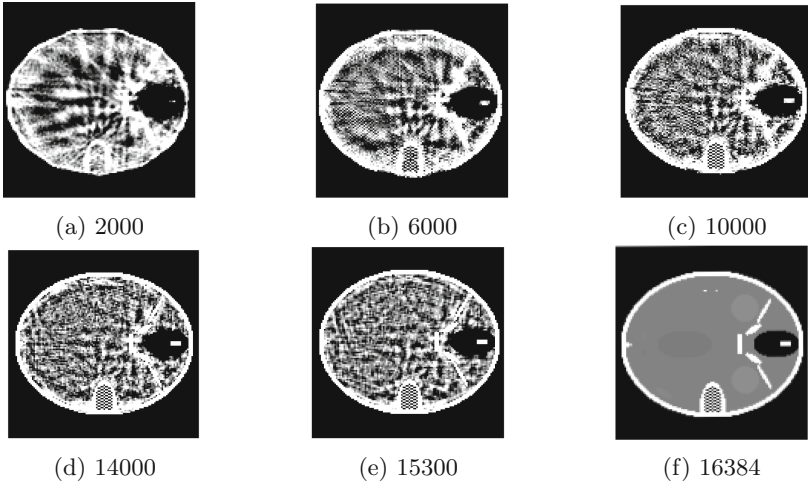
(a) 2000   (b) 6000   (c) 10000

(d) 14000   (e) 15300   (f) 16384

Fig. 4. Reconstruction varying singular values 128×128

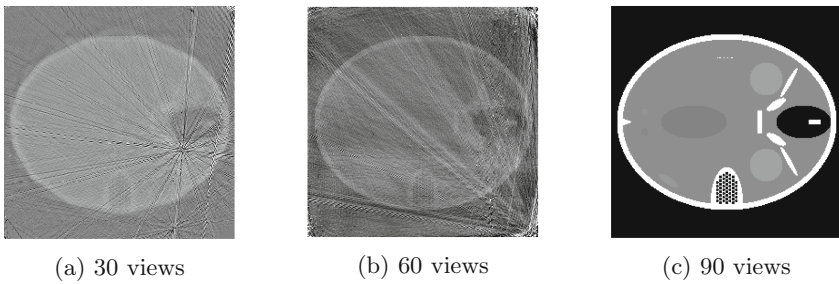(a) 30 views   (b) 60 views   (c) 90 views

Fig. 5. SPQR 256×256 reconstructions

4   Conclusion

In the present work, we have proposed two sparse matrix factorization methods
that can be used for CT image reconstruction. In this way, we simulate the
use of the pseudoinverse of the system matrix to transform the initial problem
solution into a simpler one. The main challenge of these methods is the high
consumption of memory to perform the computation, so we had to make use of
a high performance computing cluster.

In this cluster it has been possible to calculate the SPQR factorization up
to a resolution of 256×256 pixels and 90 views. In addition, the SVD up to
256×256 and 30 views. Although with the SVD method we have not obtained
satisfactory results for higher resolutions, through SPQR we have managed to
perform a reconstruction of very high quality with 90 views and full rank. This
translates into a very significant difference in the doses of x-rays to which patients
should be exposed. Taking into account that the direct methods are based on the

Nyquist theorem, for our scanner configuration, they would use significantly more views. However, we are reducing them to 90 in the case of resolution 256×256 and to 30 in lower ones.

Since the calculation of the factorizations for this type of reconstruction problem can be done before the moment of the reconstruction itself and stored, the computation times required for all cases are acceptable, the SPQR method being faster because it does not need distributed memory.

In addition, we have transformed the problem of reconstruction into a problem of matrix-vector multiplication and resolution of a triangular system in the case of the QR and a matrix-vector product in the case of the SVD. These resolutions are highly parallelizable in both CPU and GPU, which means that having the matrices stored we could accomplish the reconstructions faster than with the previous methods.

Regarding the image quality obtained, it is very satisfactory for all the cases in which the sub-matrix generated is full range, obtaining images of higher quality than those obtained by iterative methods that we have presented in previous works, such as LSQR [4]. However, the moment the sub-matrix becomes rank deficient, the reconstructed image is of very poor quality due to the lack of information. In future works, it is proposed to introduce regularization algorithms to increase the range of sub-matrices, or a filter in the image that, as with the LSQR method, are combined in a certain way that allows a more quality approximation to be made despite having an ill-conditioned problem.

In addition, it is necessary to analyze the implementation of the factorizations in order to improve the use of RAM memory, and in this way to factorize the corresponding matrix for 512×512 pixels, which is the objective resolution. For this problem, we set out to use out-of-core computing techniques that make use of the disk to avoid main memory problems.

In conclusion, we can say that we have tested the viability of the SPQR and SVD direct algebraic methods applied to CT image reconstruction. In spite of not having reconstructed images of very high quality, we have verified that with these methods it is possible to reduce the dose of radiation to a great extent if the matrix can be computed.

At this point of our work we have two resolution options: For rapid reconstructions with low dose and medium resolutions, factoring methods. For high-resolution reconstructions and slightly worse quality, the LSQR + FISTA + STF iterative method, which is slower but guarantees good results.

References

1. Andersen, A.H.: Algebraic reconstruction in CT from limited views. IEEE Trans. Med. Imag. **8**(1), 50–55 (1989)
2. Andersen, A.H., Kak, A.C.: Simultaneous algebraic reconstruction technique (SART): a superior implementation of the ART algorithm. Ultrason. Imag. **6**(1), 81–94 (1984)
3. Barrett, J.F., Keat, N.: Artifacts in CT: recognition and avoidance. RadioGraphics **24**(6), 1679–1691 (2004). https://doi.org/10.1148/rg.246045065. PMID: 15537976
4. Berry, M.W., Pulatova, S.A., Stewart, G.: Algorithm 844: computing sparse reduced-rank approximations to sparse matrices. ACM Trans. Math. Softw. (TOMS) **31**(2), 252–269 (2005)
5. Brooks, R., Chiro, G.D.: Principles of computer assisted tomography (CAT) in radiographic and radioisotopic imaging. Phys. Med. Biol. **21**(5), 689–732 (1976)
6. Chillarón, M., Vidal, V., Segrelles, D., Blanquer, I., Verdú, G.: Combining grid computing and Docker containers for the study and parametrization of CT image reconstruction methods. Proc. Comput. Sci. **108**, 1195–1204 (2017). https://doi.org/10.1016/j.procs.2017.05.065. International Conference on Computational Science, ICCS 2017, Zurich, Switzerland, 12–14 June 2017. http://www.sciencedirect.com/science/article/pii/S1877050917306038
7. Davis, T.A.: Algorithm 915, SuitesparseQR: multifrontal multithreaded rank-revealing sparse QR factorization. ACM Trans. Math. Softw. **38**(1), 8:1–8:22 (2011). https://doi.org/10.1145/2049662.2049670
8. Flores, L., Vidal, V., Verdú, G.: Iterative reconstruction from few-view projections. Proc. Comput. Sci. **51**, 703–712 (2015)
9. Flores, L.A., Vidal, V., Mayo, P., Rodenas, F., Verdú, G.: Parallel CT image reconstruction based on GPUs. Radiat. Phys. Chem. **95**, 247–250 (2014)
10. FORBILD Phantom Group. http://www.imp.uni-erlangen.de/forbild/english/results/index.htm. Accessed Sept 2016
11. Golub, G., Kahan, W.: Calculating the singular values and pseudo-inverse of a matrix. J. Soc. Ind. Appl. Math. Ser. B Numer. Anal. **2**, 205–224 (1965). https://doi.org/10.1137/0702016
12. Golub, G.H., Ortega, J.M.: Scientific Computing: An Introduction with Parallel Computing. Academic Press, Cambridge (1993). http://www.amazon.com/Scientific-Computing-An-Introduction-Parallel/dp/0122892534
13. Golub, G.H., Van Loan, C.F.: Matrix Computations. Johns Hopkins University Press, Baltimore (2013)
14. Hernandez, V., Roman, J.E., Vidal, V.: SLEPc: a scalable and flexible toolkit for the solution of eigenvalue problems. ACM Trans. Math. Softw. (TOMS) **31**(3), 351–362 (2005)
15. Hore, A., Ziou, D.: Image quality metrics: PSNR vs. SSIM. In: 2010 20th International Conference on Pattern Recognition, pp. 2366–2369. IEEE, August 2010. https://doi.org/10.1109/ICPR.2010.579, http://ieeexplore.ieee.org/document/5596999/
16. Hounsfield, G.: Computerized transverse axial scanning (tomography): part I. Description of system. Br. J. Radiol. **46**, 1016–1022 (1973)
17. Joseph, P.: An improved algorithm for reprojecting rays through pixel images. IEEE Trans. Med. Imag. **1**(3), 192–196 (1982)
18. Kak, A.C., Slaney, M.: Principles of computerized tomographic imaging. Soc. Ind. Appl. Math. (2001). https://doi.org/10.1137/1.9780898719277

19. Katsikis, V.N., Pappas, D., Petralias, A.: An improved method for the computation of the Moore-Penrose inverse matrix. Appl. Math. Comput. **217**(23), 9828–9834 (2011)
20. Maier, A., Hofmann, H.G., Berger, M., Fischer, P., Schwemmer, C., Wu, H., Müller, K., Hornegger, J., Choi, J.H., Riess, C., et al.: CONRAD—A software framework for cone-beam imaging in radiology. Med. Phys. **40**(11) (2013)
21. Nyquist, H.: Certain topics in telegraph transmission theory. Trans. Am. Inst. Electr. Eng. **47**(2), 617–644 (1928)
22. Paige, C.C., Saunders, M.A.: LSQR: an algorithm for sparse linear equations and sparse least squares. ACM Trans. Math. Softw. **8**(1), 43–71 (1982)
23. Parcero, E., Flores, L., Sánchez, M., Vidal, V., Verdú, G.: Impact of view reduction in CT on radiation dose for patients. Radiat. Phys. Chem. **137**, 173–175 (2016)
24. Radon, J.: On the determination of functions from their integral values along certain manifolds. IEEE Trans. Med. Imag. **5**(4), 170–176 (1986)