Published in final edited form as:
J Supercomput. 2009 April ; 48(1): 1–14. doi:10.1007/s11227-008-0198-9.

Parallelism of iterative CT reconstruction based on local reconstruction algorithm

Junjun Deng,
Department of Mathematics, University of Iowa, Iowa City, IA 52242, USA

Hengyong Yu,
VT-WFU School of Biomedical Engineering and Sciences, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA

Jun Ni,
Department of Radiology, University of Iowa, Iowa City, IA 52242, USA

Lihe Wang, and
Department of Mathematics, University of Iowa, Iowa City, IA 52242, USA

Ge Wang
VT-WFU School of Biomedical Engineering and Sciences, Virginia Polytechnic Institute and State University, Blacksburg, VA 24061, USA

Jun Ni: jun-ni@uiowa.edu

Abstract

An iterative algorithm is suited to reconstruct CT images from noisy or truncated projection data. However, as a disadvantage, the algorithm requires significant computational time. Although a parallel technique can be used to reduce the computational time, a large amount of communication overhead becomes an obstacle to its performance (Li et al. in J. X-Ray Sci. Technol. 13:1–10, 2005). To overcome this problem, we proposed an innovative parallel method based on the local iterative CT reconstruction algorithm (Wang et al. in Scanning 18:582–588, 1996 and IEEE Trans. Med. Imaging 15(5):657–664, 1996). The object to be reconstructed is partitioned into a number of subregions and assigned to different processing elements (PEs). Within each PE, local iterative reconstruction is performed to recover the subregion. Several numerical experiments were conducted on a high performance computing cluster. And the FORBILD head phantom (Lauritsch and Bruder http://www.imp.uni-erlangen.de/phantoms/head/head.html) was used as benchmark to measure the parallel performance. The experimental results showed that the proposed parallel algorithm significantly reduces the reconstruction time, hence achieving a high speedup and efficiency.

Keywords

Computed Tomography (CT); Image reconstruction; Iterative reconstruction; Local iterative CT reconstruction; Parallel computing; High performance computing; MPI

Correspondence to: Jun Ni, `jun-ni@uiowa.edu`.

1 Introduction

In the X-ray CT reconstruction, a cross-sectional or volumetric image of a patient is reconstructed from the projection data. There are two main approaches to perform image reconstruction, analytic, and iterative methods. Analytic methods, e.g., the FDK and the Katsevich algorithms, utilize analytic formulas to reconstruct the image of the object. The iterative methods, e.g., Algebraic Reconstruction Techniques (ART) and Expectation-Maximization (EM) [5–9], match the measured projection data with the calculated ones based on a currently approximated object density distribution, and subsequently make corrections according to the difference. This procedure is repeated until some predetermined error level or maximum iteration number has been reached.

As well-known iterative methods are superior to the analytic ones, if the projection data contains high noise or is incomplete [10]. A relatively high demand for computational time is the main drawback to use iterative methods. For example, it may take numerous hours to accomplish a single iteration to reconstruct a 3-D object with a moderate volume size from cone-beam projection data. Considering time constraints, analytic methods are favored in most tomography applications despite the limitations.

Several approaches have been developed to accelerate the computation of iterative methods. In the Ordered Subsets (OS) method, the projection data is divided into an ordered sequence of subsets (or blocks) and the image is updated after using only a subset, instead of compounding all of the projection data [11–13]. This approach is reported to be able to substantially reduce computational time while maintaining image quality [11]. In the parallel computing technology, a computational task is partitioned into multiple subtasks and the associated data is sent to different processors connected through a network. After the subtasks are completed, the results are assembled by a master processor to obtain the final result. Efforts have been made to investigate the parallel implementation of the iterative algorithms in past years [14–17] Recently, Li et al. implemented the EM algorithm and ART algorithm using the data parallelism. For a reconstruction with a grid volume 128^3 on a 16-processors PC cluster, the obtained speedup was around 9 [1].

Although the data parallelism can be used to reduce the computational time, a suffering remains due to a heavy overhead. A collective communication is required among participating processors to update the estimation of the intermediate image during each iteration. As a result, a speedup is tremendously reduced as the number of processors increases. Moreover, the approach induces a valid problem. When a computation is conducted on a low-speed network or the processors are distributed geographically, the parallel schemes are not promising at all. To resolve this problem, we proposed a parallel scheme via the local CT reconstruction algorithm developed by Wang et al. [2,3]. The parallel algorithm has the merit of reconstructing a local region of interest (ROI) without synchronizing to others processing elements (PEs). Thus, the heavy communication overhead is circumvented.

In the following sections, the local iterative CT reconstruction algorithm is first outlined. Then the corresponding parallel computing scheme is presented. Next, the performances in terms of computational times, overall speed-up, and parallel efficiency are measured and presented. Finally, we discuss some relevant issues and conclude the paper.

2 Parallel iterative reconstruction for local CT

2.1 Local iterative CT reconstruction algorithm

As illustrated in Fig. 1, the local iterative algorithm is initially proposed to address the CT reconstruction problem when projection data is incomplete [2]. Assuming a region of interest (ROI) is contained in a convex set C in the 2-D parallel beam case, the characteristic function $M(x, y)$ in C can be expressed as

$$M(x,y)=\begin{cases} 1, & (x,y)\in C,\\ 0, & \text{otherwise},\end{cases}$$

where x and y are the Cartesian coordinates. Denoting the projection profile of $M(x, y)$ as

$$P_M(\theta,t)=\int_{-R}^{R}\int_{-R}^{R}M(x,y)\delta(x\cos\theta+y\sin\theta-t)dx\,dy,$$

where $\delta(t)$ is Dirac's delta function, one can define a parameter set

$$Z=\{(\theta,t):\theta\in[0,\pi],t\in[-R,R]\text{ and }P_M(\theta,t)>0\}.$$

The projection profile along a localized parallel beam can be written as:

$$P(\theta,t)=\int_{-R}^{R}\int_{-R}^{R}f(x,y)\delta(x\cos\theta+y\sin\theta-t)dx\,dy,\qquad(\theta,t)\in Z.$$

According to Wang et al. [2,3], the local iterative reconstruction formula is given by:

$$\begin{aligned}f_{k+1}(x,y)&=\frac{f_k(x,y)}{n(x,y)}\iint_Z\delta(x\cos\theta+y\sin\theta-t)\frac{P(\theta,t)}{P_k(\theta,t)}d\theta dt\\ &=\frac{f_k(x,y)}{n(x,y)}\iint_{P_M(\theta,x\cos\theta+y\sin\theta)>0}\frac{P(\theta,x\cos\theta+y\sin\theta)}{P_k(\theta,x\cos\theta+y\sin\theta)}d\theta\\ &=f_k(x,y)g_k(x,y),\end{aligned}$$

where

$$\begin{aligned}g_k(x,y)&=\frac{1}{n(x,y)}\int_{P_M(\theta,x\cos\theta+y\sin\theta)>0}\frac{P(\theta,x\cos\theta+y\sin\theta)}{P_k(\theta,x\cos\theta+y\sin\theta)}d\theta,\\ n(x,y)&=\iint_{P_M(\theta,x\cos\theta+y\sin\theta)>0}\delta(x\cos\theta+y\sin\theta-t)dt\,d\theta,\end{aligned}$$

and

$$P_k(\theta,t)=\int_{-R}^{R}\int_{-R}^{R}f_k(x,y)\delta(x\cos\theta+y\sin\theta-t)dx\,dy,\qquad(\theta,t)\in Z$$

is the reprojected data based on current image estimate $f_k(x, y)$.

NIH-PA Author Manuscript

NIH-PA Author Manuscript

In the cone-beam geometry, the projection is considered as a blurred three dimensional function:

$$P(\vec{\alpha})=\int f(\vec{X}_p)\Delta(\vec{X}_p,\vec{\alpha})d\vec{X}_p,$$

where $\vec{\alpha}=(\beta, p, \xi)'$, β denotes the X-ray source rotation angle, (p, ξ) specifies the detector position,

$$\Delta(\vec{X}_p,\vec{\alpha})=\Delta(\vec{X}_p,\vec{X}_s,\vec{X}_d)=\delta\left(\frac{x_p-x_s}{x_d-x_s}-\frac{y_p-y_s}{y_d-y_s}\right)\delta\left(\frac{x_p-x_s}{x_d-x_s}-\frac{z_p-z_s}{z_d-z_s}\right),$$

$\vec{X}_p \equiv (x_p, y_p, z_p)'$, $\vec{X}_s \equiv (x_s, y_s, z_s)'$, $\vec{X}_d \equiv (x_d, y_d, z_d)'$, are vectors for specimen, source, and detector coordinates, respectively. The associated iterative formula is:

$$f_{k+1}(\vec{X}_p)=\frac{f_k(\vec{X}_p)}{H_0(\vec{X}_p)}\int\Delta(\vec{X}_p,\vec{\alpha})\frac{P(\vec{\alpha})}{P_k(\vec{\alpha})}d\vec{\alpha}=f_k(\vec{X}_p)g_k(\vec{X}_p),$$

where

$$g_k(\vec{X}_p)=\frac{1}{H_0(\vec{X}_p)}\int\Delta(\vec{X}_p,\vec{\alpha})\frac{P(\vec{x}_d)}{P_k(\vec{x}_d)}d\vec{\alpha},$$

and

$$\begin{aligned}H_0(\vec{X}_p)&=\int\Delta(\vec{X}_p,\vec{\alpha})d\vec{\alpha}\\&=\iiint\delta\left(\tfrac{x_p-x_s}{x_d-x_s}-\tfrac{y_p-y_s}{y_d-y_s}\right)\delta\left(\tfrac{x_p-x_s}{x_d-x_s}-\tfrac{z_p-z_s}{z_d-z_s}\right)dp\,d\xi d\beta.\end{aligned}$$

Geometrically, $P_k(\vec{X}_p)$ is a synthesized cone beam projection based on the current estimate $f_k(\vec{X}_p)$, $g_k(\vec{X}_p)$ is the overall correction factor computed by backprojecting the ratios of measured and synthesized projections, and $H_0(\vec{X}_p)$ is the weight compensating for cone beam divergence.

The local iterative algorithm can be considered as a generalized EM-type algorithm. If the set *C* further represents the whole object scanned, the algorithm is virtually identical to the conventional EM algorithm. In general, the set *C* is a nontrivial part of the object, and the algorithm can accurately recover high-frequency information in the set *C*, while faithfully providing low-frequency information outside of it.

2.2 Parallel reconstruction using iterative local CT algorithm

This section presents the strategy for reconstructing a 3-D object in parallel, by using the local iterative CT reconstruction algorithm. Conventionally, to parallelize the computation of an iterative algorithm, the projection data is first partitioned into several groups and sent to different PEs. Then each PE uses the projection data to complete the reconstruction. After each iteration, the PEs exchange the current estimation with the other PEs and continue the next iteration, until either a predetermined error is tolerated or maximum iteration is reached. The approach reconstructs images identically to the associated sequential algorithm. However, as mentioned in the introduction section, it suffers from a heavy communication overhead. Hence, the performance is compromised if a large number of processors are used.

The proposed local reconstruction algorithm allows boosting the performance by reducing heavy communication overhead. And the algorithm can reconstruct the local region of interest C with high accuracy. Therefore, one can partition a 3-D object data into multiple sub-ROIs, and assign each sub-ROI to a single PE. Within each PE, the local iterative CT reconstruction algorithm is deployed to perform concrete reconstruction by regarding the assigned sub-ROI as the set C in Wang's algorithm. Once all the PEs have accomplished their tasks, a master node assemble the sub-ROIs images collected from all PEs into the final reconstruction. In practice, to ensemble the final image, a collective communication operation "MPI_Gatherv" is used to gather the results from slave nodes in order. Finally, the master node could either save the result to disk or send it to a remote user. Figure 2 illustrates the whole flowchart.

It can be observed that unlike the conventional approaches, there is no communication among the PEs during each iteration. The reason is the reconstruction of a sub-ROI on the assigned PE is fully independent of others. The only communication time used is to collect images of all the sub-ROIs from all the PEs once. In this way, the communication overhead among processing elements (PEs) is eliminated and the parallel performance significantly increases. The approach maximizes the efficiency because a single sub-ROI result can be achieved independently while other sub-ROIs results are under computation. This property is favorable in a distributed environment.

Since both the size and the position of a sub-ROI in the whole ROI influence the reconstruction time, load imbalance is a more sensitive issue than that in the conventional approaches. Without careful consideration, the parallel performance would be compromised by the load imbalance. Due to the symmetry of the scanning locus in X and Y direction, one can partition data evenly in both directions. In the primary study, we partitioned the ROI into 2 by 2 equal grids in the X–Y plane. In the Z direction, the partition is more complicated. It can be verified that the computational load on each PE is roughly proportional to the number of X-rays that intersect with the associated sub-ROI. Since the X-ray source emits a cone beam consisting of equal number X-rays at all positions on the circular scanning locus in our simulation model, partitioning evenly along the Z-direction seems to be a good choice. However, when the X-ray source starts from the bottom and ends at the top of the phantom, such as a spiral scanning locus, some of the X-rays emitted from these positions do not intersect with the phantom. Such X-rays have little contribution to the whole computation and are removed before the iteration begins. Consequently, the computation for the X-rays from these positions is less than that for other positions. Figure 3 gives an illustration of this situation. It's impossible to give a universal partitioning criterion so that the parallel computing is synchronized perfectly. Nevertheless, we could manually adjust the partitioning ratio for a much smaller group of projection data. Since the CT scanning geometry is not changed, if the load imbalance is resolved for the smaller case, then the larger cases are also settled.

3 Experiments

To demonstrate the feasibility of the parallel iterative local CT algorithm, numerical experiments were designed and conducted using the FORBILD head phantom [4]. The parallel algorithm was implemented on a PC cluster at Medical Imaging High Performance

Computing Lab (MIHPC Lab) at the University of Iowa. The cluster has 16 nodes, each consisting of two 64-bit AMD Opteron processors with 4 GB memory. Message Passing Interface (MPI), a parallel library, was used to perform message passing (process of data communication) among the PEs. The program was written in C, and compiled by the Porland Group's c compiler.

As an example, we chose the practical spiral cone beam scanning geometry in our simulation. The geometrical parameters were summarized in the Table 1. A planar detector was used to collect the projection data. Two cases with different projection data and reconstruction matrix size were used.

Table 2 gives the results of the measured computational time with respect to the number of PEs. In both cases, the computational time is significantly decreased as the number of PEs increases.

To examine the performance of the proposed parallel algorithm, we computed the two standard benchmarks, speedup S_p and efficiency η, which are defined as

$$S_p = \frac{T_s}{T_{np}}, \text{ and } \eta = \frac{S_p}{n_p}.$$

Here n_p is the number of processors, T_s is the total execution time when one processor is used, and T_{np} is the total parallel execution time when n processors are used.

The speedup and efficiency were computed from the Table 1. And the results were presented in Table 3 and plotted in Fig. 4. In Fig. 4(a), the speedup linearly increases with the increase of the number of processors. This is a considerable advantage over the conventional parallel iterative algorithms, where the speedup increases initially and then decreases due to an inevitably large amount of communication overhead [1]. This behavior is very promising to achieve high performance in a large-scale system with more computer processors. Another interesting observation is that the speedup curves for the two cases are close to each other, regardless of the difference of data size (projection data and the reconstruction matrix).

To further accelerate the parallel computing, a special strategy, reconstructing the region outside of the ROI with lower resolution, can be applied by taking advantage of the local iterative CT reconstruction algorithm. Since the algorithm only recovers the low frequency information for the regions outside the sub-ROIs and we are only interested in the reconstruction inside the sub-ROIs, we can tolerate lower resolution outside of the sub-ROI while reconstructing high-resolution inside the sub-ROI. This is feasible since the iterative CT algorithm is implemented in a ray-tracing manner. Along the ray, we use larger step size to trace forward and backward when the ray is outside of the sub-ROI, and keep the step size when the ray is inside of the sub-ROI. Therefore, the computational time for the outside region can be reduced and the total computational time can be decreased as well. Upon this idea, we conducted several experiments. Table 4 gives the computational time when using different resolutions for the inner and outside sub-ROI regions. The computational time is further reduced comparing with the previous homogeneous resolution approach. The speedup and efficiency in Fig. 4 clearly verify this point.

In order to show the applicability of the parallel algorithm to preserve the reconstruction quality, some typical reconstructed image slices were presented in Fig. 5 and representative profiles were plotted in Fig. 6. Comparing these results with the one using sequential algorithm, a congruency can be seen from the Figs. 5 and 6, indicating the image quality was quite stable regarding the different ways of partitioning the ROI. This was consistent with what Wang et al. mentioned that the local iterative approach will naturally reconstruct an actual image when sufficient projection data is available, or will produce an optimal image [2]. It could also be observed that the image quality was well maintained when moderately lower resolution for the region outside of the sub-ROI was used.

As we could observe, although the speedup consistently increases, it continues to show a gap between the ideal speedup—the straight line with a unit slope. The reason is that although the PE reconstructs the sub-ROI independently, it also recovers the low-frequency information outside of this sub-ROI, which introduces the redundant computation in the parallel scheme. Therefore, theoretically, the parallel reconstruction using the local iterative CT algorithm won't obtain the linear speedup or unit efficiency.

4 Discussion and conclusion

Although a good parallel performance was achieved, the load imbalance caused by the ROI partitioning was not solved thoroughly. As we mentioned in Sect. 2, the partitioning criterion in Z direction was based on the test for smaller data set, which was more or less imprecise. Besides, it is not convenient to adjust the partitioning ratio when the number of processors is large. More handy methods need to be exploited to solve this problem thoroughly.

Another concern is about the quality of the reconstructed image. It is clear that the parallel algorithm was not identical to its sequential prototype. Therefore, there was a bright spot in the center of the slice, which was the boundary of the sub-ROIs. To remove this, one could append more layers to the boundary of the sub-ROIs when reconstructing and retrieving only the central parts to resemble the final result.

In addition to the ROI partitioning, the heterogeneous resolution is also a key factor that determines the load of each PE, and thus affects the speedup and load imbalance potentially. Generally speaking, the lower resolution for the region outside ROI, the higher speedup could be expected. However, the increase in speed for each PE might not be identical since the partition itself is not homogeneous. Furthermore, there should be a tradeoff between it and the image quality, since the coarser resolution in the outside region still has impact on the ROI. As a result, a balanced point needs to be carefully chosen so as to achieve an optimal result.

In conclusion, a parallel computing strategy based on local iterative CT reconstruction algorithm was investigated in this paper. To perform the parallel computing, a ROI was partitioned to sub-ROIs and each sub-ROI was assigned to a PE. On each PE, the local iterative CT reconstruction algorithm was used to conduct the reconstruction. Then the master node collected all the sub-ROIs from the worker nodes to assemble the image. As a result, the computational time was greatly reduced and high speedup was achieved. A special strategy using inhomogeneous resolution was taken to further speedup the computation while the image quality was preserved. Future research should include investigating the impact of different partition methods on the performance of the parallel algorithm, a more detailed investigation into the effect of inhomogeneous resolution on the speedup and image quality and the study on how to removing the bright spot on the boundary of the sub-ROIs while preserving the parallel performance.

Acknowledgments

The project was supported by National Health Institute (NIH/NIBIB) grants EB001685, EB002667, and EB006412-01. The authors would like to thank Mark Fleckenstein, University of Michigan, for editorial help in preparing the manuscript. Thanks also go to Research Services of Information Technology Services, the University of Iowa, for their administrative and computing supports.

References

1. Li X, Ni J, Wang G. Parallel iterative cone beam CT image reconstruction on a PC cluster. J X-Ray Sci Technol 2005;13:1–10.
2. Wang G, Snyder DL, Vannier MW. Local computed tomography via iterative deblurring. Scanning 1996;18:582–588. [PubMed: 8946772]
3. Wang G, Snyder DL, O'Sullivan JA. Iterative deblurring for CT metal artifact reduction. IEEE Trans Med Imaging 1996;15(5):657–664. [PubMed: 18215947]
4. Lauritsch, G.; Bruder, H. Head phantom technical report. http://www.imp.uni-erlangen.de/phantoms/head/head.html
5. Andersen AH. Algebraic reconstruction in CT from limited views. IEEE Trans Med Imaging 1989;8:50–55. [PubMed: 18230499]
6. Andersen AH, Kak AC. Simultaneous algebraic reconstruction technique (SART): A superior implementation of the ART algorithm. Ultrasonic Imaging 1984;6:81–94. [PubMed: 6548059]
7. Dempster AP, Laird NM, Rubin DB. Maximum likelihood from incomplete data via the EM algorithm. J R Stat Soc (B) 1977;39:1–39.
8. Shepp LA, Valdi Y. Maximum likelihood reconstruction for emission tomography. IEEE Trans Med Imaging 1982;1(2):113–122. [PubMed: 18238264]
9. Lange K, Carson R. EM reconstruction algorithms for emission and transmission tomography. J Comput Assist Tomogr 1984;8(2):302–316.
10. Leahy RM, Qi J. Statistical approaches in quantitative positron emission tomography. Stat Comput 2000;10:147–165.
11. Hudson HM, Larkin RS. Accelerated image reconstruction using ordered subsets of projection data. IEEE Trans Med Imaging 1994;13:601–609. [PubMed: 18218538]
12. Kamphuis C, Beekman FJ. Accelerated iterative transmission CT reconstruction using an ordered subsets convex algorithm. IEEE Trans Med Imaging 1998;17:1101–1105. [PubMed: 10048870]
13. Kole JS, Beekman FJ. Evaluation of the ordered subset convex algorithm for cone-beam CT. Phys Med Biol 2005;50:613–623. [PubMed: 15773623]
14. Miller M, Butler C. 3-D maximum a posteriori estimation for single photon emission computed tomography on massively-parallel computers. IEEE Trans Med Imaging 1993;12:560–565. [PubMed: 18218449]
15. Chen CM, Lee SY, Cho ZH. A parallel implementation of 3-D CT image reconstruction on hypercube multiprocessor. IEEE Trans Nucl Sci 1990;37(3):1333–1346.
16. Chen CM, Lee SY. On parallelizing the EM algorithm for PET image reconstruction. IEEE Trans Parallel Distributed Syst 1994;5(8)
17. Johnson, C.; Sofer, A. Proc of 7th IEEE symp front mass parallel computing. IEEE Computer Society Press; 1999. A data-parallel algorithm for iterative tomographic image reconstruction.

Biographies

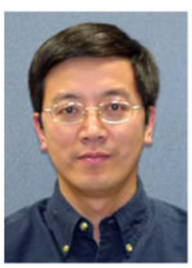
Junjun Deng received his B.S. (2000) and M.S. (2003) in mathematics from Peking University, Beijing. Currently he is a Ph.D. candidate in the program in mathematics and computational sciences, the University of Iowa, USA. His interests include computed tomography, medical image processing and parallel computing.

Hengyong Yu earned his B.S. degrees in information science & technology (1998), computational mathematics (1998), and Ph.D. in information & telecommunication engineering (2003) from Xi’an Jiaotong University, China. He was Instructor and Associate Professor with the College of Telecommunication Engineering, Hangzhou Dianzi University, from July 2003 to Sept. 2004. During Sept. 2004 and Nov. 2006, I was postdoctoral fellow and Associate Research Scientist with the Department of Radiology, University of Iowa, Iowa City, IA. During Sept. 2004 and Nov. 2006, I was postdoctoral fellow and Associate Research Scientist with the Department of Radiology, University of Iowa, Iowa City, IA. During Sept. 2004 and Nov. 2006, he was postdoctoral fellow and Associate Research Scientist with the Department of Radiology, University of Iowa, Iowa

City, IA. Currently, he is a Research Scientist and the Associate Director of the CT Laboratory, VT-WFU School of Biomedical Engineering and Science, Virginia Tech, Blacksburg, VA. His interests include computed tomography and medical image processing. He has authored or coauthored more than 50 peer-reviewed journal papers. He serves as the Editorial Board Member of Signal Processing, Guest Editor of the International Journal of Biomedical Imaging and Guest Associate Editor for Medical Physics. He is a senior member of the IEEE and member of the Chinese Institute of Electronics. In 2005, he was honored for an outstanding doctorial dissertation by Xi'an Jiaotong University, and received the first prize for the best natural science paper from the Association of Science & Technology of Zhejiang Province, China.

Jun Ni received his B.S. from Harbin Engineering University, China, in 1982, M.S. from Shanghai Jiao Tong University (SJTU), China, in 1984, and his Ph.D. from the University of Iowa (UI), USA, in 1991. He had a research associate/lecturer position in SJTU. He worked as postdoctoral associate at UI and Purdue University from 1992 to 1994. Since 1994, he worked at UI as a senior computing consultant, associate research scientist, research scientist, and the director of Scientific Computing (high performance computing and grid computing). He has been an adjunct assistant and associate professor in the Department of Computer Science and the Department of Mechanical & Industrial Engineering at UI. Currently, he is an Associate Professor and director in the Department of Radiology, College of Medicine, Associate Professor of Biomedical Engineering and Mechanical Engineering at UI, and Adjunct Associate professor in Computer Science. He is the director of Medical Imaging HPC Lab (MiHi) and HPC Nanotechnology Lab (HPCNano) at UI. He has over 100 peer-reviewed conference and journal papers, 30 edited books. He is an editor-in-chief, associate and guest editor of more than 20 journals. He is a member of IEEE, ASME, SMII, AASA, FAS, and RSNA.

NIH-PA Author Manuscript



Lihe Wang received his B.S. from Peking University, China, in 1983, M.S. from the University of Chicago, in 1986, and his Ph.D. from New York University 1989. He worked as instructor and assistant professor at Princeton University from 1989 to 1993. Since 1993, he worked at University of Iowa as Associate Professor and full professor. He has been as associate professor in UCLA from 1995 to 1997 and member of IAS in 2002. He was also a Sloan Fellow in 1994.

NIH-PA Author Manuscript

Ge Wang (S'90–M'92–SM'00–F'03) received B.E. in electrical engineering from Xidian University, Xian, China, in 1982, M.S. in remote sensing from Graduate School of Academia Sinica, Beijing, China, in 1985, and M.S. and Ph.D. in electrical and computer engineering from State University of New York, Buffalo, in 1991 and 1992. He was Instructor and Assistant Professor with Department of Electrical Engineering, Graduate School of Academia Sinica in 1984–1988, Instructor and Assistant Professor with Mallinckrodt Institute of Radiology, Washington University, St. Louis, MO, in 1992–1996. He was Associate Professor with University of Iowa from 1997–2002, and then Professor with Departments of Radiology, Biomedical Engineering, Mathematics, Civil Engineering, Electrical and Computer Engineering, and Director of the Center for X-Ray and Optical

Tomography, University of Iowa. Currently, he is Director of the Biomedical Imaging Division and Samuel Reynolds Pritchard professor, WFU-VT School of Biomedical Engineering and Sciences at Virginia Polytechnic Institute and State University, Blacksburg, VA. His interests include computed tomography, bioluminescence tomography, and systems medicine. He and his coauthors have published over 400 journal articles and conference papers, including the first paper on spiral/helical cone-beam CT, the first paper on bioluminescence tomography, and the first paper on interior tomography. He is the founding Editor-in-Chief for *International Journal of Biomedical Imaging*, and Associate Editors for *IEEE Trans. Medical Imaging* and *Medical Physics*. He is an IEEE Fellow, SPIE Fellow, and an AIMBE Fellow. He is also recognized by a number of awards for academic achievements.

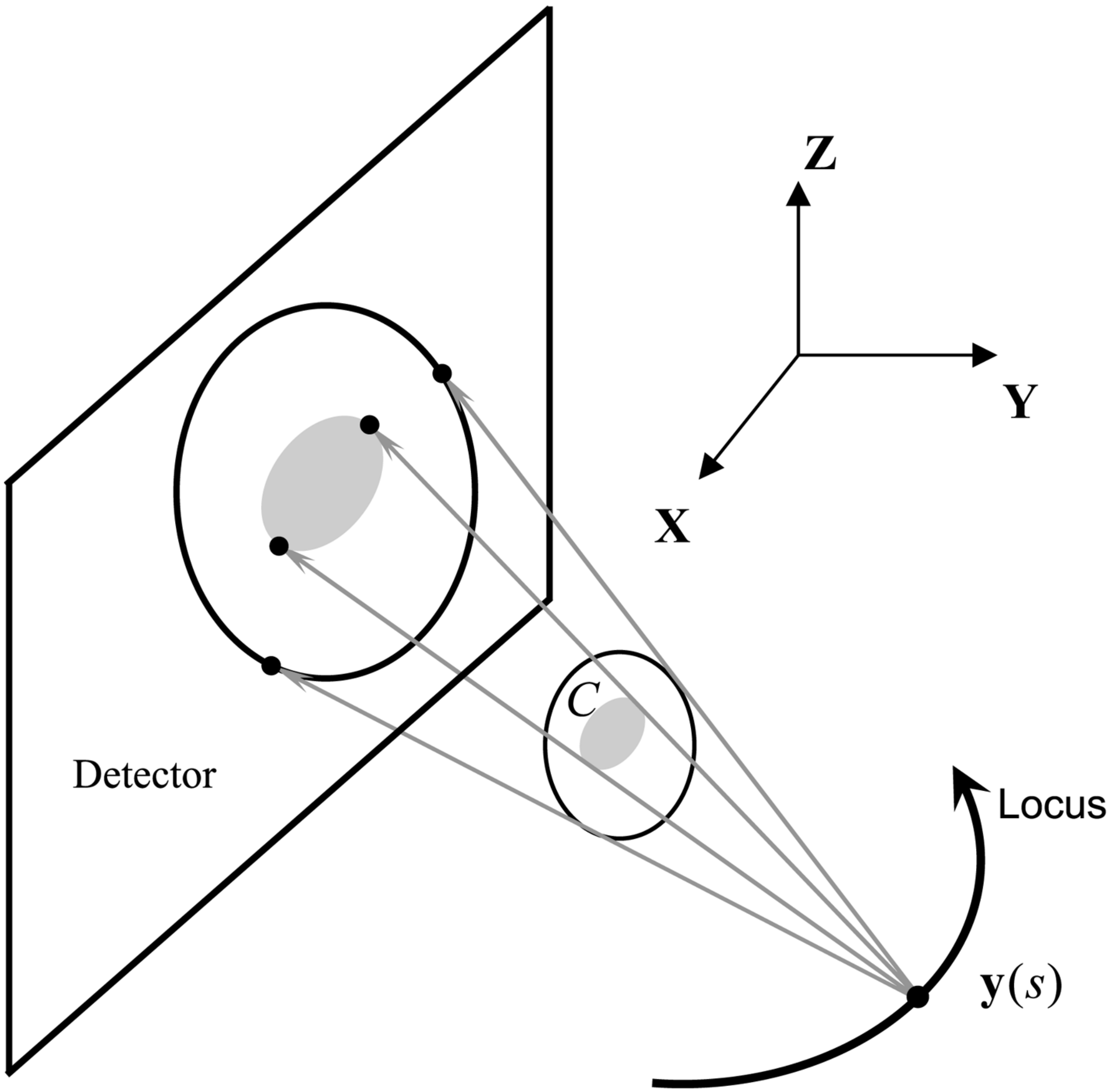


Fig. 1.
Geometrical illustration of the cone-beam CT system

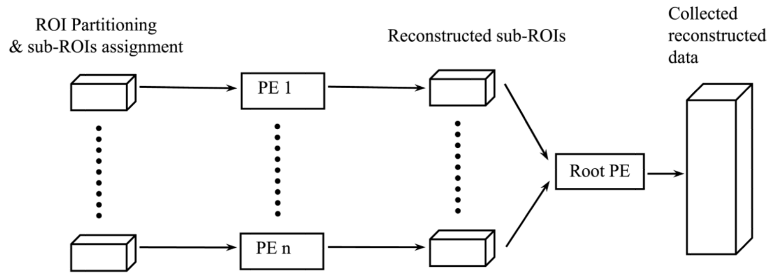


Fig. 2.
The flowchart of the parallel algorithm

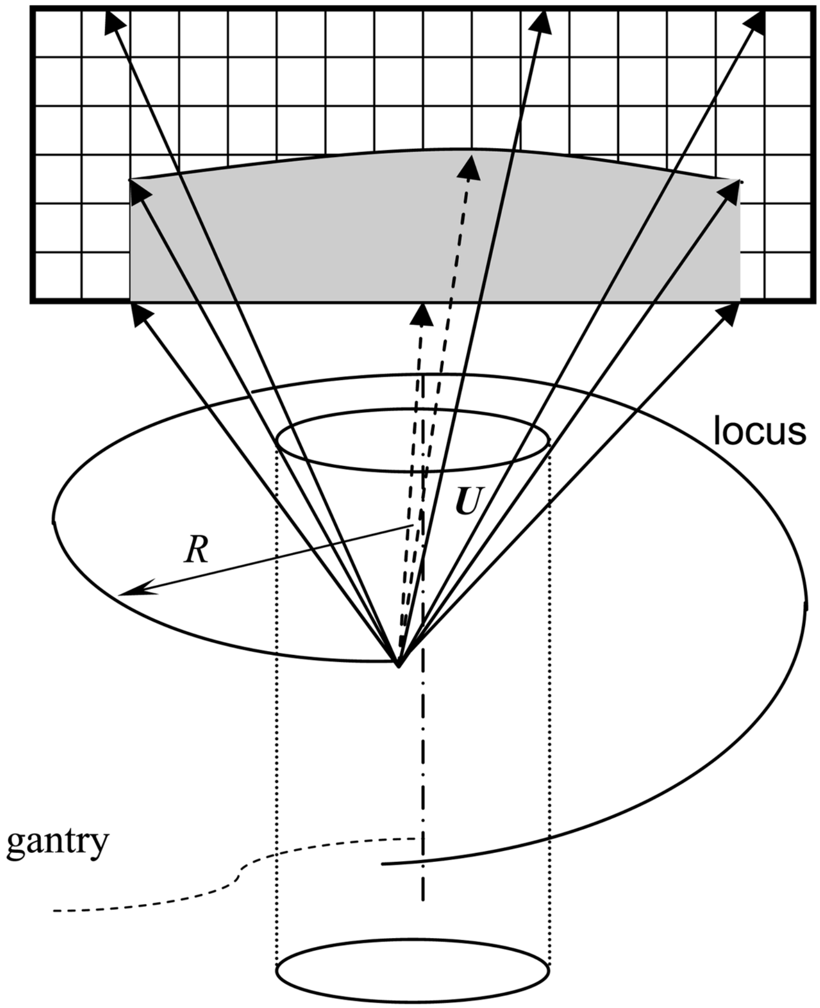


Fig. 3.
Illustration where only part of the cone beam X-ray intersect with the object

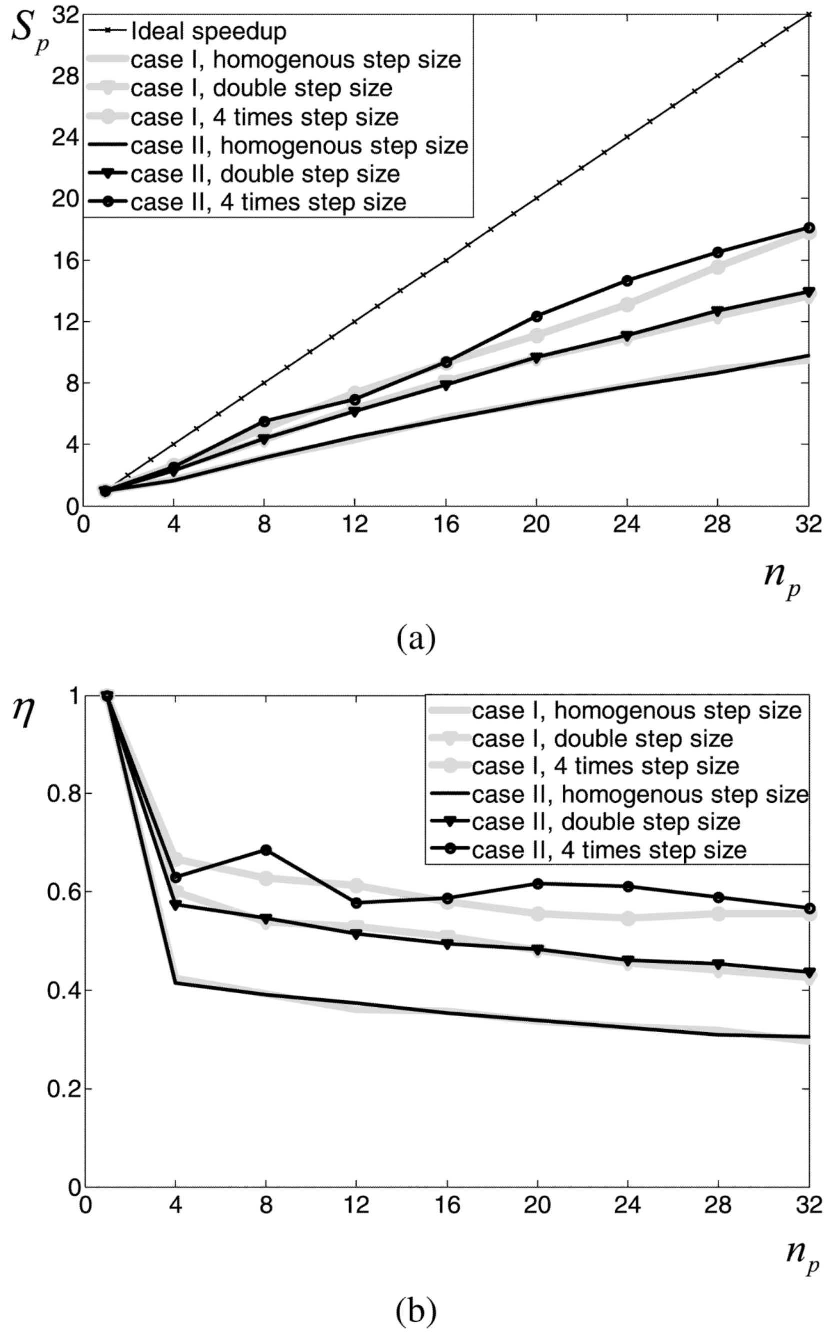


Fig. 4.
Comparison of the (**a**) speedup, and (**b**) efficiency of the parallel iterative algorithm

NIH-PA Author Manuscript

NIH-PA Author Manuscript

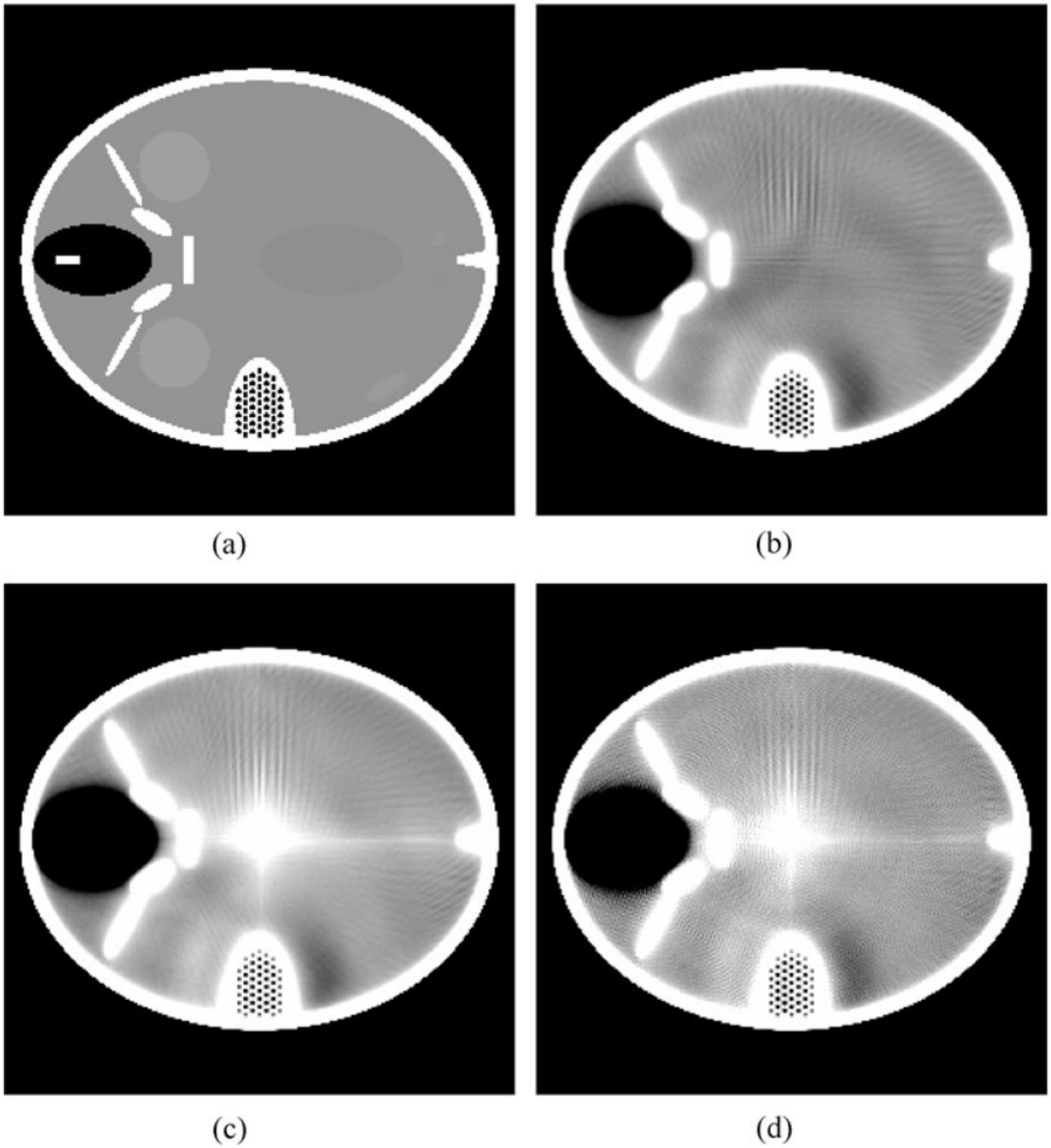


Fig. 5.
Representative slices of reconstructed 256^3 volume. (**a**) Original phantom, (**b**) sequential EM algorithm, (**c**) homogenous step size, (**d**) double step size. Displaying window for call cases is [0.95,1.15], where the value in the range is linearly rescaled to [0, 255]

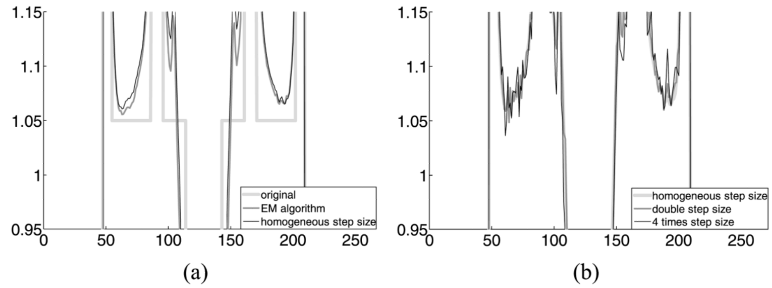


Fig. 6.

Representative profiles of reconstructed slices. (**a**) The profiles of the original phantom, reconstruction result of EM algorithm, and reconstruction results of the parallel algorithm, respectively. (**b**) The profiles of the reconstruction results when using homogeneous step size, double step size for outside sub-ROIs region, and 4 times step size for the outside sub-ROIs region, respectively

NIH-PA Author Manuscript
NIH-PA Author Manuscript

Table 1

Parameters of the spiral cone beam geometry

	Case I	**Case II**
Scanning radius (cm)	64	64
Source to detector distance (cm)	128	128
Helical pitch (cm)	12.8	6.4
Object radius (cm)	12.8	12.8
Detector size (width, height)	28.41 × 22.53	28.41 × 22.53
Number of projections per turn	96	192
Number of detector cells	128 × 64	512 × 256
Reconstruction matrix	128^3	256^3

Table 2

Reconstruction time with different number of processors (NP)

NP	1	4	8	12	16	20	24	28	32
Case I	1310	771	418	302	229	194	168	147	138
Case II	157448	94732	50500	35125	27878	23294	20308	18190	16113

Note: The unit of time is second

NIH-PA Author Manuscript
NIH-PA Author Manuscript

Table 3

Speedup and efficiency with different number of processors (NP)

NP	1	4	8	12	16	20	24	28	32
Speedup (Case I)	1.00	1.70	3.13	4.34	5.72	6.75	7.80	8.91	9.49
Speedup (Case II)	1.00	1.66	3.12	4.48	5.65	6.76	7.75	8.66	9.77
Efficiency (Case I)	1.00	0.42	0.39	0.36	0.36	0.34	0.33	0.32	0.30
Efficiency (Case II)	1.00	0.42	0.39	0.37	0.35	0.34	0.32	0.31	0.31

Table 4

Computational time with different NP when using heterogeneous resolution

NP	1	4	8	12	16	20	24	28	32
Case I, double step size	1310	545	304	206	161	136	120	106	96
Case I, 4 times step size	1310	467	248	169	134	112	95	80	70
Case II, double step size	157448	68444	36066	25523	19913	16299	14211	12398	11376
Case II, 4 times step size	157448	62444	28716	22738	16765	12744	10717	9544	8685

Note: The unit of time is second. Here, we compare the results when double step size and 4 times step size are used for the region outside the sub-ROIs when local reconstruction is carried out on the PEs

Table 5

Speedup and efficiency with different number of processors (NP), inhomogeneous cases

NP	1	4	8	12	16	20	24	28	32
Speedup (Case I, double step size)	1.00	2.40	4.31	6.36	8.14	9.63	10.92	12.36	13.65
Speedup (Case I, 4 times step size)	1.00	2.67	5.02	7.37	9.29	11.12	13.11	15.56	17.79
Speedup (Case II, double step size)	1.00	2.30	4.37	6.17	7.91	9.66	11.08	12.70	13.97
Speedup (Case II, 4 times step size)	1.00	2.52	5.48	6.92	9.39	12.35	14.69	16.50	18.13
Efficiency (Case I, double step size)	1.00	0.60	0.54	0.53	0.51	0.48	0.45	0.44	0.43
Efficiency (Case I, 4 times step size)	1.00	0.67	0.63	0.61	0.58	0.56	0.55	0.56	0.56
Efficiency (Case II, double step size)	1.00	0.58	0.55	0.51	0.49	0.48	0.46	0.45	0.44
Efficiency (Case II, 4 times step size)	1.00	0.63	0.69	0.58	0.59	0.62	0.61	0.59	0.57