Asian Agricultural Research 2013, 5(12): 90 – 95

Parallelized LEDAPS method for Remote Sensing Preprocessing Based on MPI

Xionghua CHEN, Xu ZHANG*, Ying GUO, Yong MA, Yanchen YANG

Research Institute of Forest Resources Information Techniques, Chinese Academy of Forestry, Beijing 100091, China

Abstract Based on Landsat image, the Landsat Ecosystem Disturbance Adaptive Processing System (LEDAPS) uses radiation change detection method for image processing and offers the surface reflectivity products for ecosystem carbon sequestration and carbon reserves. As the accumulation of massive remote sensing data especially for the Landsat image, the traditional serial LEDAPS for image processing has a long cycle that make a lot of difficulties in practical application. For this problem, this paper design a high performance parallel LEDAPS processing method based on MPI. The results not only aimed to improve the calculation speed and save computing time, but also considered the load balance between the flexibly extended computing nodes. Results show that the highest speed ratio of parallelized LEDAPS reached 7.37 when the number of MPI process is 8. It effectively improves the ability of LEDAPS to handle massive remote sensing data and reduces the forest carbon stocks calculation cycle by using the remote sensing images.

Key words Forest carbon stock, LEDAPS, Landsat image preprocessing, MPI, Parallel computing

The research of forest biomass and carbon stock is a hot issue of science emerging in the context of global warming, and global climate change is one of the most serious environmental problems currently facing mankind[1]. Remote sensing estimation method is used to obtain a variety of vegetation state parameters by means of remote sensing, combined with ground surveys, so as to complete vegetation spatial classification and time series analysis; then it can be used to analyze the temporal and spatial distribution and dynamic change of forest ecosystem carbon, and estimate the impact of carbon stock of a large area of forest ecosystem and land use change on carbon stock[2]. As a kind of remote sensing estimation method, LEDAPS (Landsat Ecosystem Disturbance Adaptive Processing Systems) carries out calibration, cloud mask, accurate registration, and the preprocessing of orthorectification and atmospheric correction on Landsat image, to provide information on forest disturbance. At the same time, it can provide surface reflectivity products for a variety of ecosystem researches based on Landsat image[3]. With the rapid development of remote sensing technology and the accumulation of massive remote sensing data[4] especially for the Landsat image, LEDAPS has a exponentially growing demand for computing resources, and the traditional serial LEDAPS processing remote sensing image gradually cannot keep up with the generation of remote sensing image, which has put forward higher requirements on the computing resources and computing power of remote sensing processing computing platform. In recent years, the high performance parallel computing has experienced the rapid development and become the key to maintaining the competitive advantage of science and technology[5]. MPI is a message passing interface standard, for developing the parallel programs based on message passing[6]. Compared with the traditional serial computing, parallel computing can accomplish the same computing tasks with shorter time and less input, so using parallel computing to carry out task processing is more efficient, but the cost is much lower[7]. Using MPI to build high performance parallel computing platform has opened up a new way for LEDAPS to process remote sensing image data.

This article uses MPI to carry out coarse – grained data parallel processing and transformation on LEDAPS. The master – slave model is adopted to conduct task allocation on the parallel Landsat image data. Meanwhile, based on the cyclic division Landsat image data set method, we carry out the systematic load balancing, to achieve the scalability of computing nodes, based on substantially increasing the computing speed, and saving computing time.

1 Introduction of LEDAPS

LEDAPS (The Landsat Ecosystem Disturbance Adaptive Processing System) is developed by a research group under NASA (the U.S. National Aeronautics and Space Administration). It is a part of the North American carbon research program, and also a part of the U.S. global change carbon cycle science research program[8]. The system uses the radiation change detection method for mapping forest disturbance information, to get forest distribution, recovery information, and then apply it to the research of forest biomass and carbon stock. In order to better carry out radiation change detection, LEDAPS software first conducts a series of preprocessing steps on the original Landsat images inputted, including calibration, cloud detection, accurate registration, orthorectification and atmospheric correction.

During calibration, this software uses MODTRAN solar output model to correct the original image spectral values to the corresponding spectral values at the top of the atmosphere (TOA);

Received: August 3, 2013 Accepted: October 10, 2013

Supported by Special Project of Fundamental Research Funds for Central Public Welfare Research Institutes (IFRIT201104).

* Corresponding author. Email: zhangxu@caf.ac.cn

then the cloud detection and cloud mask are completed; using AROP package algorithm, the accurate registration and orthorectification of remote sensing image are carried out[9]; finally using 6S (the Second Simulation of the Satellite Signal in the Solar Spectrum) radiative transfer method, atmospheric correction is carried out on the images after calibration and cloud mask[10], to convert the spectral values of image to surface reflectivity values and form the corresponding surface reflectivity products[11]. Such image preprocessing occupies most of the processing time in the entire system of LEDAPS. After the image preprocessing, we can map the forest disturbance information. Currently, there are two kinds of algorithms for forest disturbance mapping contained in the LEDAPS software. The first algorithm is to use analyze the interference index to identify interference pixel and restoration pixel for mapping. The difference between the first algorithm and the second algorithm is that the second algorithm does not directly identify the pixels, but directly extract the vegetation structure parameter needed by the carbon modeling for mapping[3]. After the above process is completed, the resulting forest disturbance information mapping product can provide a wealth of information, such as the disturbed forest types and disturbance time. At the same time, this product can also provide forest restoration information, such as the stage of forest restoration. The specific processes of LEDAPS software is shown in Fig. 1.

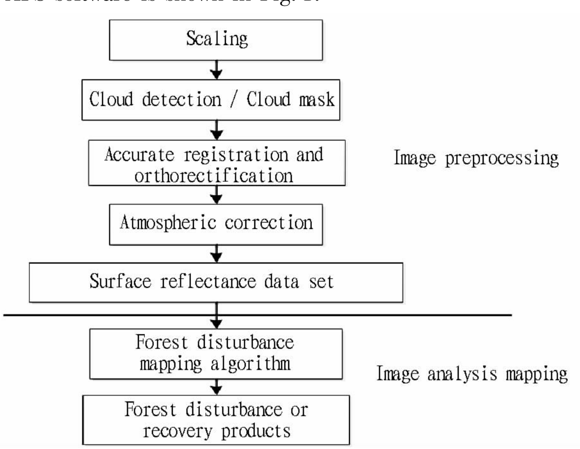


Fig. 1 The processing processes of LEDAPS software

2 Introduction of MPI

Message Passing Interface (MPI) is a standardized and portable message – passing system designed by a group of researchers from academia and industry to function on a wide variety of parallel computers[12]. The standard defines the syntax and semantics of a core of library routines useful to a wide range of users writing portable message – passing programs in FORTRAN or the C programming language. There are several well – tested and efficient implementations of MPI, including some that are free or in the public domain. These fostered the development of a parallel software industry, and there encouraged development of portable and scalable large – scale parallel applications. Currently the framework adopted by most MPI parallel clusters is the single – layer structure[13]. The structure consists of a master node and a plurality of slave nodes, and the communication links are established between the master node and each slave node to achieve polymerization communication. The point to point communication can be achieved between slave nodes[14]. The main function of the master node includes interaction with the users, broadcasting and distributing messages to the slave nodes, collecting and summarizing the messages sent by the slave node. The main function of the slave nodes is computing, receiving the various control commands and data sent by the master node, carrying out according execution in its node machine according to the control command, and then returning the results executed to the master node.

3 Parallelization method for LEDAPS image preprocessing

In the whole LEDAPS system, for a Landsat image with a size of about 100 MB, the preprocessing will take about 20 minutes, accounting for about 90% of the overall system processing. Therefore, this study mainly carries out parallel transformation of the image preprocessing section. In the image preprocessing section, the time is mainly spent on the atmospheric correction using 6S radiative transfer method. The principle of 6S atmospheric correction is radiative transfer equation. The basic monochromatic radiative transfer equation in the plane – parallel atmosphere[15] is as follows: $\frac{dI\nu(1)}{d\tau} = I\nu(\tau, \mu, \varphi) - J\nu(\tau, \mu, \varphi)$

where τ is vertical optical thickness; μ is the cosine of the zenith angle of the path that the light passes through; φ is the orientation angle of path relative to the sun.

During the processing, it involves a lot of numerical computing, and the computing is intensive and time – consuming. The preprocessing mode of LEDAPS is single Landsat data source entry, and there is no interactive between different processing data. For the processing mechanism, it has strong data – level parallel processing capability, and it is easy to expand, so this study uses coarse – grained data parallel mode to carry out preprocessing of LEDAPS image and achieve parallel transformation.

3.1 Preprocessing parallel transformation of LEDAPS image based on MPI The specific transformation method is to transform the original single – channel process based on MPI into the multi – channel process using the master – slave mode. Among them, the master node is used to control data distribution and process scheduling, and the slave node is used to carry out conventional preprocessing of LEDAPS image. The work to be done by the master process is as follows: reading the original Landsat image data, generating the Landsat data list assigned to each slave process, sending the Landsat data list to the corresponding slave node, and outputting the processed data. The work to be done by the slave process is as follows: establishing the input parameter file for the assigned Landsat data, calibrating, cloud detection, accurate registration, orthorectification, and atmospheric correction. The parallel processing flow diagram is shown in Fig. 2.

(i) The master process reads the Landsat data set, and makes the data shared. (ii) According to the number of slave process, the master process carries out the balanced scheduling on the Landsat data set, to generate the data list files processed by the slave process according to the process number, and send the MPI_SEND function to each slave process. (iii) Through MPI_RECV function, the slave process receives the data list files sent by the master process. (iv) The slave process opens data list files, and according to the data list, read the processed Landsat image data from the sharing data of the master process. (v) The original Landsat data is decoded. (vi) The slave process calls LEDAPS program, and establishes the input parameter file for the atmospheric correction of Landsat data. (vii) According to the MODTRAN solar output model, the slave process carries out calibration, to generate TOA reflectivity images. (viii) The slave process carries out cloud detection and cloud mask of Landsat remote sensing data. This process will produce temporary processing files with the same name. Due to the adoption of MPI parallel processing, in order to avoid the error of the same name covering, the parallel processing process carries out the detection and judgement of the temporary files. If there is the temporary file with the same name, then this slave process will delay for 10s, then re – detect and re – judge until there are no temporary files with the same name. The cloud detection processing of this slave process is conducted again. (ix) The slave process uses AROP program algorithm to carry out the accurate registration and orthorectification processing of remote sensing image. (x) The slave process uses 6S radiative transfer method to carry out atmospheric correction on the images after calibration and cloud detection. (xi) The slave process outputs the processing results after atmospheric correction to the main program. (xii) Judging whether it is the end of the data list files. If not, the Landsat image preprocessing from (v) to (xi) is repeated; if yes, exit the program, and complete the processing work of this slave process.

3.2 Load equalization During the parallel processing, when using MPI to carry out task allocation for remote sensing image, there will be the phenomenon of load imbalance. So this research uses the way of circular assignment of input Landsat data set, to make the data allocated balanced to each slave process and ensure the balanced load of system, reduce the overall system processing time and improve the parallel processing efficiency. Before load balancing, the image data allocation method is as follows:

If $N / P = M$, then M image data are allocated to the first $P-1$ processes one by one, and the last process is allocated with $N-M \times (P-1)$ image data (N is the input data set, and P is the number of processes).

The improvement method for load balancing is as follows:

If $N \bmod P \neq 0 = R$, then the first R processes are allocated with image data one by one, and the last $P-R$ processes are allocated with image data one by one (N is the input data set, and P is the number of processes).

3.3 Task mapping In the division of coarse – grained data, the slave nodes are independent, and there is not much data communication between them. The system is easy to extend, but there is a lot of data communication between the master node and slave node. In order to improve the efficiency of parallel computing and reduce the data communication between the master node and slave node, this research uses the task mapping approach and adopts the form of file list, to establish mapping between the data classified and slave process, so that the communication between the master process node and slave process node is minimized.

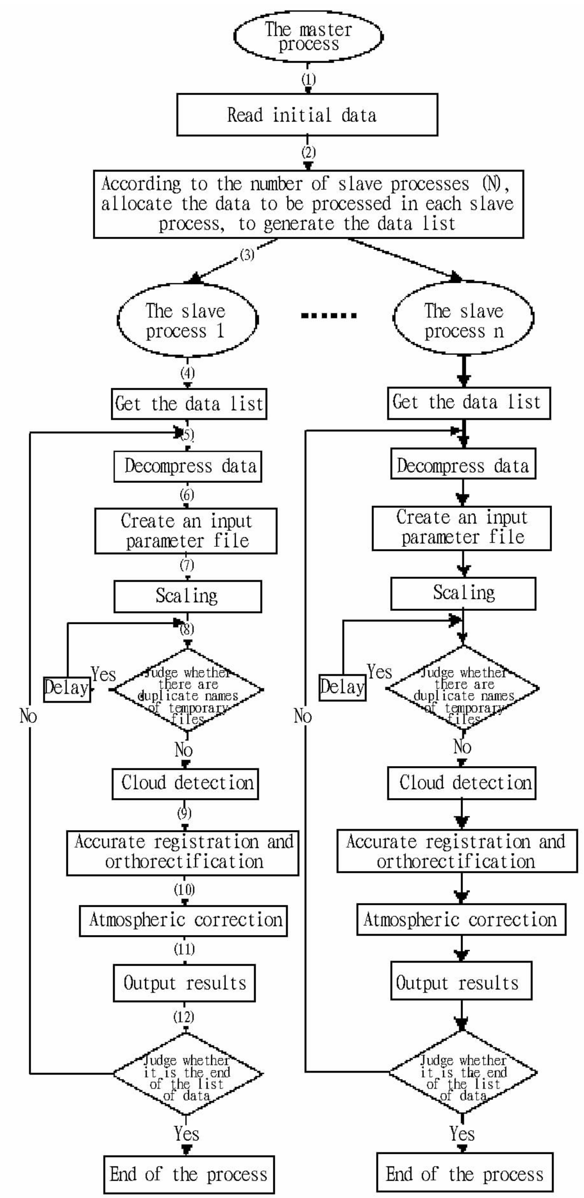


Fig.2 The parallel flow chart

The specific mapping method is as follows: In the master process nodes, after the circular division of the input original Landsat data set through the way of load balancing, the file list to be processed that each slave process node corresponds to is generated, and through the MPI_RECV function, the file list is sent to the corresponding slave process node. After the slave process node obtains the corresponding file list, it only need to obtain the corresponding processed Landsat data set files from the master process node, thereby reducing the data communication of Landsat data set in the network, and improving the performance of parallel computing.

4 Testing and result analysis

The parallel transformation platform uses the high performance computing cluster of the Chinese Academy of Forestry as the operating platform. The platform deploys 48 computing nodes, to be connected via gigabit Ethernet switch, using HP ProLiant BL460c G6 blade server. The operating system of platform is CentOS 5.4, the MPI compiler is mpich2, and the compilation environment is C. The testing data are the Landsat TM images of the three northeastern provinces in China, and the size of each TM image is about 150M. This paper compared the execution time of serial program based on the single CPU and the parallel program with a number of MPI processes based on high performance cluster running platform for processing 10 TM images, 30 TM images and 50 TM images, respectively with LEDAPS. We also compared the performance of load balancing for MPI multi – processing. To ensure the reliability of the results, the method used averaged time as the unit of each testing, and the statistical results are shown in Table 1 (The unit of running time is minute.).

Table 1 Performance test results

Number of images	Serial program	4 MPI parallel computing nodes						8 MPI parallel computing nodes					
		Before load balancing			After load balancing			Before load balancing			After load balancing		
		Iteration frequency	Run time	Speedup	Iteration frequency	Run time	Speedup	Iteration frequency	Run time	Speedup	Iteration frequency	Run time	Speedup
10	193	4	82	2.35	3	59	3.27	3	61	3.16	2	37	5.21
30	577	9	187	3.09	8	158	3.65	9	189	3.05	4	83	6.95
50	984	14	273	3.60	13	257	3.82	8	159	6.19	7	133	7.37

Through the experimental results in Table 1, it can be seen that with the increasing of the TM data, the time spent on the execution of the serial LEDAPS is significantly increased. For the 8 MPI parallel computing nodes before load balancing, 6 image data are allocated to the first 7 computing modes, and 8 images are allocated to the last computing node. The maximum number of iterations is 8, the overall running time is 159 minutes, and the speedup is 6.19. After load balancing, 7 image data are allocated to the first 2 computing nodes, and 6 image data are allocated to the last 6 computing nodes. The maximum number of iterations is 7, the overall running time is 133 minutes, and the speedup is 7.37. After load balancing, the number of iterations of MPI parallel computing is reduced and the overall running time is also reduced. In the study, we use speedup as another indicator for assessing the parallel computing performance. From Fig. 3, 4, it can be seen that when the amount of data involved in the processing does not change, with the increase of the computing nodes (for example, during the increase of computing nodes from 2 to 10), there is a decreased linear relationship between the processing time and the number of computing nodes. Speedup (i. e., the ratio of serial execution time to parallel execution time) is increased with the increase of the computing nodes in a linear manner. This shows that the parallelized LEDAPS has good scalability.

To test the processing performance of LEDAPS MPI parallel program for the massive remote sensing image data, we also compare the average time of processing different size of Landsat TM image based on the fixed 20 MPI parallel computing nodes. The results are shown in Fig. 5. The parallel processing time is the running time after load balancing.

From Fig. 5, it can be seen that with the increase in the amount of TM remote sensing data, there is linear increase in the processing time of MPI parallel program. When 300 TM images which is about 45G processed, the execution time is 384 minutes. In order to measure the processing performance of parallelized LEDAPS for massive data, we also introduce the probability of " accelerated efficiency", namely the ratio of time spent on the preprocessing of LEDAPS when there is one iteration of MPI computing nodes. The processing results are shown in Fig. 6. Compare to 10 images, the parallel acceleration efficiency of 300 TM images is not reduced, but increased. Because in the multi – process processing, as the amount of data increases and the number of iterations increases, the delay time of the temporary files with the same name is offset mutually, shortening the processing time of whole process and thereby increasing the efficiency of parallel processing. To verify the parallel computing results of LEDAPS program, we select the ETM + image in the Northeast in April 2010 for processing analysis, as shown in Fig. 7.

Fig. 7 offers the original image, TOA image and the image after atmospheric correction. From these figures, we can find that before atmospheric correction, the edges of terrain are blurred, and after atmospheric correction, the effects of the atmosphere are eliminated and the contrast is enhanced, so the original appearance of the terrain and its features are significantly restored, and the edges of terrain become clear.

For the quantitative evaluation of correction effect, we read 50 vegetation samples from the images before atmospheric correction and the images after atmospheric correction, respectively, and calculate the surface reflectance without atmospheric correction and the surface reflectance after atmospheric correction, as shown in Table 2. Before correction, due to atmospheric effects, the minimum value of each band is greater than that of the corrected image, and after correction, the standard deviation of each band is greater than that of image before correction. The corrected image has richer information and the image enhancement effect is significant.

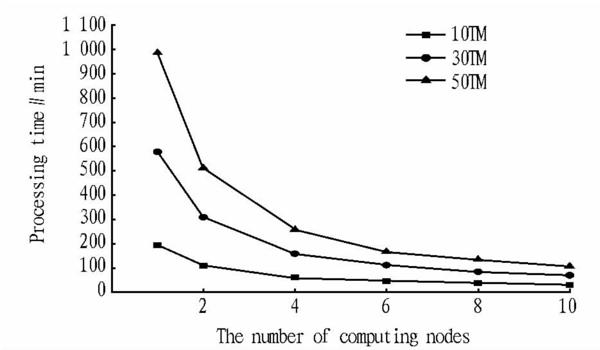


Fig. 3 The parallel processing effect

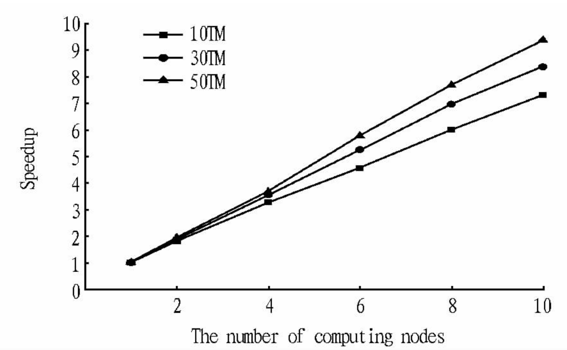


Fig. 4 The parallel speedup effect

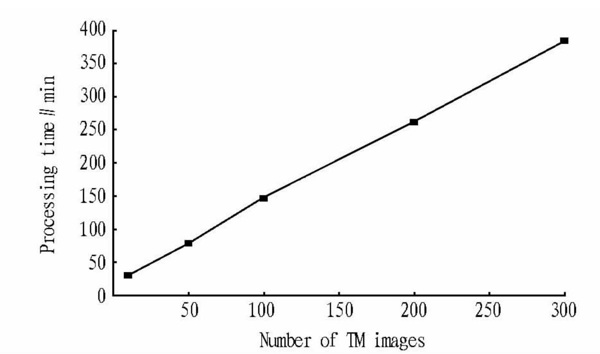


Fig. 5 MPI parallel processing of different orders of magnitude

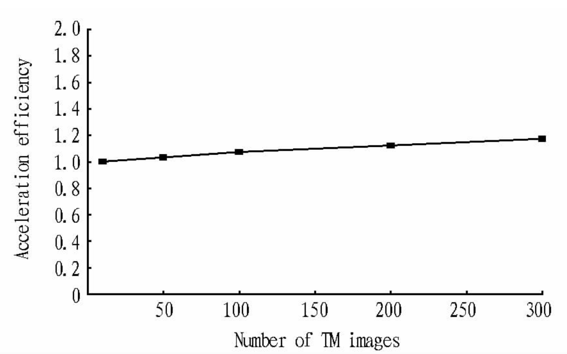
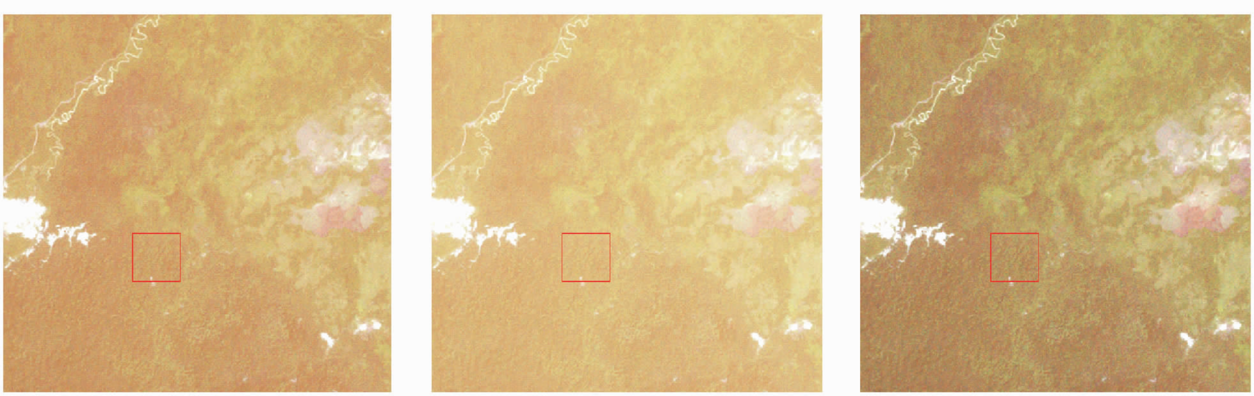


Fig. 6 Speedup efficiency of different orders of magnitude



(a) Original image (b) TOA image (c) Image after atmospheric correction

Fig. 7 Original image, TOA image and the image after atmospheric correction

Table 2 Comparison of the statistical characteristics of image in the study area before and after atmospheric correction

Wave band	Before atmospheric correction				After atmospheric correction			
	Minimum	Maximum	Mean	Standard deviation	Minimum	Maximum	Mean	Standard deviation
B1	0.084 7	0.097 6	0.091 1	0.002 1	0.042 3	0.060 0	0.051 2	0.002 8
B2	0.061 2	0.093 2	0.075 8	0.004 1	0.031 8	0.070 9	0.049 5	0.005 1
B3	0.046 5	0.068 7	0.054 9	0.003 9	0.028 8	0.054 8	0.038 7	0.004 6
B4	0.137 2	0.320 4	0.224 7	0.028 8	0.136 9	0.341 0	0.236 5	0.031 6
B5	0.077 6	0.190 8	0.122 7	0.023 5	0.072 7	0.209 1	0.133 8	0.026 0
B7	0.048 9	0.094 5	0.057 0	0.012 3	0.031 8	0.106 0	0.063 5	0.014 0

5 Conclusions

The forest is regarded as the largest terrestrial carbon storage library and the most economical carbon absorption unit. The research of carbon sink of forest has attracted the attention of the international community increasingly[16]. LEDAPS based on Landsat image uses the radiation change detection method, to obtain forest vegetation information, and provide surface reflectivity products for estimating the carbon stock of a large area of forest ecosystem. With the development of remote sensing technology, the traditional serial LEDAPS preprocessing can not meet the needs of massive remote sensing image. On the basis of in – depth analysis of the principles and features of LEDAPS processing, this paper proposes a parallel computing method for LEDAPS massive image preprocessing based on MPI, considering the problem of time – consuming process of original serial image preprocessing of LEDAPS. The method uses a master-slave mode focuses on the irrelevance between the Landsat data processing, and carries out circular classification of Landsat data, to achieve data parallelization and at the same time achieve the load balancing processing of the task. The experimental results show that when 8 MPI processes run the program, the speedup after load balancing is 7.37. With the increase of computing nodes, the MPI speedup is increased linearly. At the same time, the test results also show that LEDAPS after transformation has good scalability, and is able to minimize the communication between nodes on the basis of load balancing. This parallel method is easy to use, robust and scalable. On the basis of substantially increasing the computing speed and saving the computing time, it realizes the scalability of computing nodes. This will help to apply LEDAPS to the processing of Landsat massive data, form large-scale application of corresponding surface reflectivity products, and accelerate the application of remote sensing technology to forest biomass and carbon stock.

References

[1] XU TS. Study on forest biomass & carbon storage estimation based on remote sensing information[J]. Forest Inventory and Planning, 2008, 33(3): 11 – 13. (in Chinese).

[2] YANG HX, WU B, ZHANG JT, *et al.* Progress of research into carbon fixation and storage of forest ecosystems[J]. Journal of Beijing Normal University(Natural Science), 2005, 41(2): 172 – 176. (in Chinese).

[3] Wolfe R, Masek J, Saleous N, *et al.* LEDAPS: mapping North American disturbance from the Landsat record. In: Geoscience and Remote Sensing Symposium[C]. IGARSS'04 Proceedings. IEEE International, 2004.

[4] LIU HY, LI ES, ZHANG ZH. Study on image parallel OPTA algorithm based on thinning cluster[J]. Engineering of Surveying and Mapping, 2011, 20(20): 35 – 38. (in Chinese).

[5] LU YT. Trend study of extreme large scale parallel computing systems[J]. Computer Engineering & Science, 2012, 34(8): 17 – 23. (in Chinese).

[6] XIAO HL, WANG LS. Investigation on parallel computing algorithm of DNS using pseudo-spectral method based on MPI[J]. Computer Engineering and Applications, 2012, 48(4): 54 – 55. (in Chinese).

[7] HUANG LX, SHAO JL, CUI YL, *et al.* Study on the parallel algorithm for numerical simulation of groundwater based on the MPI[J]. Journal of Anhui Agricultural Sciences, 2012, 40(12): 7356 – 7358. (in Chinese).

[8] Wolfe R, Hall F, Goward S *et al.* Assessing North American forest disturbance from the landsat archive[J]. Geoscience and Remote Sensing Symposium, 2007. IGARSS 2007. IEEE International, 2007, 5294 – 5297.

[9] LI AN, JIANG JG, BIAN JH, *et al.* Experiment and accuracy analysis of automated registration and orthorectification for Landsatlike images based on AROP[J]. Remote Sensing Technology and Application, 2012, 27(1): 23 – 32. (in Chinese).

[10] JU J, Roy D P, Vermote E, *et al.* Continental-scale validation of MODIS-based and LEDAPS Landsat ETM^{+} atmospheric correction methods[J]. Remote Sensing of Environment. 2012(122): 175 – 184.

[11] Jeffrey G. Masek, Eric F. Vermote, Nazmi E. Saleous *et al.* A Landsat surface reflectance data set for North America, 1990 – 2000[J]. Geoscience and Remote Sensing Letters, 2006, 3(1): 68 – 72.

[12] ZHANG LB, CHI XB, MO ZY. Introduction to parallel computing[M]. Beijing: Tsinghua University Press, 2006. (in Chinese).

[13] WANG W, ZHANG JM. Design and implementation of MPI parallel computing system based on tree structure[J]. Application of Electronic Technique, 2011, 37(12): 126 – 129. (in Chinese).

[14] CHEN CP. The parallel technologies (PaCT – 2003)[C]. Nizhni Novgorod, Russia. 2003.

[15] WANG Y, FAN WY, YANG AL. The application of 6S models for radiative transfer for correcting remote sensing image in Harbin City[J]. Geomatics & Spatial Information Technology, 2008, 31(2): 16 – 19. (in Chinese).

[16] WANG CM, WANG RN, LIN ZL. Carbon sequestration estimation by different tree species and reforestation types in forest ecosystem[J]. Ecology and Environmental Sciences, 2010, 19(10): 2501 – 2505. (in Chinese).

(From page 89)

[45] LUO GY, ZHU L, JI TJ, *et al.* Effects of different P concentration and aeration parttens on growth of algae in freshwaters [J]. Journal of Chongqing University(Natural Science Edition), 2007, 30(2): 86 – 88, 106. (in Chinese).

[46] WANG YZ, YANG CJ, CHEN XD, *et al.* Effects of different water dynamical condition on algae succession in eutrophication process of landscaping water body[J]. The Administration and Technique, 2011, 23(2): 23 – 27. (in Chinese).

[47] DONG LL. The effect of submerged macrophytes together with sediments on the circulation of phosphorus of eutrophic lakes[D]. Tianjin: Tianjin University, 2007: 12 – 16. (in Chinese).