

The Analysis Meteorological Satellite Software Based on Principal Component

Manyun Lin, Xiangang Zhao, Cunqun Fan^{*}, Lizi Xie, Lan Wei

National Satellite Meteorological Centre, China Meteorological Administration, Beijing, China
{linmy, zhaoxg, fancq, xielz, weilan}@cma.gov.cn

Keywords: meteorological satellite software; principal component; feature data.

Abstract: How to provide reasonable hardware resources and improve the efficiency of software is paid more and more attention. In this paper, a set of software classification method based on software operating characteristics is proposed. The method uses software run-time resource consumption to describe the software running characteristics. Principal component analysis (PCA) is used to reduce the dimension of software running feature data and to interpret software characteristic information. Simulation results show that the proposed method can optimize the allocation of software hardware resources and improve the efficiency of software operation.

1. Introduction

With the increase of meteorological satellite observation level and the rich variety of remote sensing products, meteorological satellite remote sensing products are more and more widely used. National Satellite Meteorological Center Fengyun meteorological satellite ground application system deal with a large number of satellite observations in real time every day. It have put forward high requirements on the timeliness and reliability for ground application system data processing. At the same time, it challenges the design and operation of these applications to support the work of IT platform. How to fully understand the resource requirements of various types of data processing software and make effective use of IT resources has become an urgent problem in the field of meteorological satellite ground application system.

Various types of meteorological satellite data processing prototype software in National Satellite Meteorological Center are the crystallization of countless scientist's hard work for several years. With the development of remote sensing instruments and the development of remote sensing technology, the ground meteorological software is constantly enriched and renewed. In engineering construction, these prototyping software become an important component of Fengyun meteorological satellite ground application system after engineering. It should be necessary to establish the detection and evaluation methods, after the engineering data processing software and the use of hardware resources to assess the rationality. Fengyun meteorological satellite ground application system has a large number of data processing software, so classification of software resources and the use of the characteristics of its operation is the basis for carrying out evaluation work.

Experimental data used in this paper is from the collected data on the operation of Fengyun-3C data processing software. First of all, we collected the original software running feature data,

processing feature extraction, to better express the characteristics of the software. Secondly, principal component analysis (PCA) was used to analyze the operational characteristics of the collected data, and the principal components were extracted and their features were described. Then the clustering analysis is carried out by using the processed software characteristic data to realize the classification of meteorological software, such as computing-intensive, memory-intensive, I/O-intensive and network-intensive. Finally, based on the results of PCA, the characteristics of each type of software are described, which provides basic data and basis for further work, such as software resource consumption rationality analysis, software operation rationality evaluation, optimization of hardware and software systems, and provides scientific decision data support for future hardware and software platform planning and configuration of new projects.

2. Extraction and Processing of Software Running Feature

2.1 Software and hardware environment overview.

The object of this paper are the 182 sets of polar orbiting meteorological satellite data processing software of the 12 categories of instruments for the Fengyun-3C satellite ground application system. Hardware resources, including 6 IBM minicomputers, de-tailed configuration in Table 1.

Tables 1. Hardware Configuration.

serv_name	cpu_hz	cpu_type	cpu_core	real_mem	vir_mem
coss1-3c	4.4Ghz	POWER7	32	186G	62G
coss2-3c	4.4Ghz	POWER7	32	186G	62G
pgs1-3c	4.4Ghz	POWER7	22	134G	52G
pgs2-3c	4.4Ghz	POWER7	22	134G	52G
pgs6-3c	4.2Ghz	POWER7	12	250G	78G
pgs7-3c	4.2Ghz	POWER7	12	250G	78G

2.2 Maintaining the integrity of the specifications.

Software operating characteristic data acquisition range contained 182 sets of polar orbit meteorological satellite data processing software for 12 kinds of instruments. Polar orbit meteorological satellites carried remote sensing instrument. Its mode of operation is to collect data on a regular basis and download the collected data to the ground station. The software needs to run multiple times per day (each run is called a track). Data acquisition environment is the simulation environment and acquisition time is 4 days. The collection method for operating characteristics of the software is to force each weather processing software running in serial (the actual environment running is in parallel), so that each software can get sufficient hardware resources and give full play to software performance. Software operating characteristics data acquisition types included CPU, system, process, and job level data, with CPU-level and system-level acquisition cycles of 1 second. Job-level data acquisition fields are the main software start time, end time and the located server. System-level data acquisition fields are CPU system and disk wait for usage, CPU idle usage, memory usage, virtual memory usage, disk read and write rates, network receive and send rate. CPU-level data acquisition field has the core CPU system utilization and idle utilization.

2.3 Characterization of operational characteristics.

Software feature analysis needs to express the operating characteristics of the software as much as

possible, and ultimately to express the operating characteristics of each software through a vector [1]. Characterization of the software running characteristics need to consider from two aspects: (1) time-series characteristics of software operation; (2) to eliminate differences between the platforms and the resource consumption of the system (only consider the resources consumed by the software itself).

The time-series features of the software running are represented by peak, mean and summation of resource consumption [2]. Eliminating platform differences requires the conversion of resource usage to usage. The consumption of resources of the software itself needs to throw away the occupied resources of the system. To this end, we carried out based on the parameters of the collected information and software running on the server information, synthesis of new feature parameters. Specific treatment is as follows:

Software running time:

$$t = t_{start} - t_{end} \quad (1)$$

CPU user calculation:

$$cpu_{am} = (100 - cpu_{rat_{syswa}} - cpu_{rat_{idle}}) * cpu_{core} * cpu_{hz} \quad (2)$$

CPU calculation total:

$$cpu_{sum} = \sum_{i=0}^t cpu_{am}_i \quad (3)$$

CPU calculated peak:

$$cpu_{max} = \max(cpu_{am}_i) \quad (4)$$

Memory usage:

$$mem = mem_{rat} * real_{mem} \quad (5)$$

Virtual memory usage:

$$swap = swap_{rat} * vir_{mem} \quad (6)$$

Disk read:

$$disk_{read}_{sum} = \sum_{i=0}^t disk_{read}_i \quad (7)$$

Disk write:

$$disk_{write}_{sum} = \sum_{i=0}^t disk_{write}_i \quad (8)$$

Network receiving:

$$net_{rec}_{sum} = \sum_{i=0}^t net_{rec}_i \quad (9)$$

Network sending:

$$net_{send}_{sum} = \sum_{i=0}^t net_{send}_i \quad (10)$$

Through the above conversion, the software's each track operating characteristic data is transferred into a vector, and then we calculate the average value of the software multi-track running characteristics, finally formatted a 182×14 data matrix of the original operating characteristics.

3. Principal Component Analysis

Principal Component Analysis (PCA) is a statistical method [5]. Through orthogonal transformation to a group of variables may be related to the conversion of a group of linearly unrelated variables, the group of variables after transformation is called the principal component. The results of principal component analysis are mainly dependent on the correlation between indicators. If the correlation is very strong, the results of principal component analysis will be very good, otherwise it is poor [6].

According to the paper [4,7,8], when ρ (cumulative %) $\geq 0.8 \sim 0.9$, we can use the first five principal components instead of the original 14 operating characteristics, and retain the original 14 operating characteristics contain the main information, the first five principal components are called public influence factors.

According to the analysis in Table 3, the cumulative values of the four principal components of 1,2,3,4 are 78.471%, which can represent the main factors of the original matrix. In the process of running, the expression of the variable is not the original variable, but the standardized variable, such as the first principal component.

Tables 2. Explain the total variance.

Ingredients	Initial eigenvalue			Extract the square load		
	Total	Variance %	Cumulative %	Total	Variance %	Cumulative %
1	4.948	35.346	35.346	4.948	35.346	35.346
2	2.724	19.457	54.803	2.724	19.457	54.803
3	1.815	12.965	67.768	1.815	12.965	67.768
4	1.498	10.702	78.471	1.498	10.702	78.471
5	0.813	5.808	84.278			
6	0.666	4.759	89.037			

In Table 3, it can be found that the main components in the first category are mainly related to run time and disk read and write resources. The second category are mainly related to network resources and CPI. The third category is mainly related to computing resources. And the fourth category are related to memory and cache.

Tables 3. Main ingredient.

1	2	3	4
Running time	CPI average	Computational complexity	The maximum cache size
Disk read total	Network Sending Peak	Computational peak value	The maximum memory
Disk read peak	Network Receiving Total		
Disk write total	Network Receiving Peak		
Disk write peak			
Network Sending Total			

4. Summary

In this paper, principal component analysis algorithm are used to classify the software in the meteorological field, and the software classification in the meteorological field is solved. At the same time, the characteristics of each kind of software are analyzed. Using the results of the classification,

the software scheduling algorithm is further analyzed to improve the hardware utilization and reduce the soft-ware waiting time.

Acknowledgements

The work presented in this study is supported by National High-tech R&D Program (2011AA12A104).

References

- [1] CHEN Zhuang, LUO Chengcheng. Application of an improved K-means algorithm in anomaly detection [J]. Journal of Chongqing University of Technology: Natural Science, 2015 (5): 66-70
- [2] Yong S U, Liu Q, Peng L M, et al. Application of Principal Components Analysis and Cluster Analysis in Evaluation of Blending Module[J]. Tobacco Science & Technology, 2005,30 (2): 365-369.
- [3] LI Zuo-Yong, DING Jing, PENG Li-Hong. Principles and methods of environmental qual-ity assessment [M]. Chemical Industry Press, 2004,5 (1)
- [4] JIA Rui-yu, SONG Jian-lin.K-means optimal cluster number determination method based on clustering center optimization [J] .Microelectronics & Computer, 2016,33 (5): 62-66
- [5] YIN Cheng-xiang ZHANG Hong-jun ZHANG Rui QI Xiu-li WANG Bin An Improved K-Means Algorithm [J] .Computer Technology and Development, 2014,24 (10): 30-33.
- [6] Kuang F, Zhang S, Jin Z, et al. A novel SVM by combining kernel principal component analysis and improved chaotic particle swarm optimization for intrusion detection[J]. Soft Computing, 2015, 19(5):1187-1199.
- [7] Rocha H, Li W, Hahn A. Principal Component Regression for Fitting Wing Weight Data of Subsonic Transports[J]. Journal of Aircraft, 2015, 43(6):1925-1936.
- [8] Liu W, Zhang H, Tao D, et al. Large-scale paralleled sparse principal component analysis[J]. Multimedia Tools and Applications, 2016, 75(3):1-73.