Available online www.jocpr.com

Journal of Chemical and Pharmaceutical Research, 2014, 6(6):2743-2750

Research Article

ISSN : 0975-7384
CODEN(USA) : JCPRC5

A map-task view Generation strategy Based on Rough Set Theory

Yiyi Xu*[1.2], Peihe Tang[1] and ZeKun Tang[1]

[1]College of Computer Engineering, Guangxi University of Science and Technology, Liuzhou, 545006, China
[2]School of Information Engineering, Wuhan University of Technology, Hubei, 430070, China

ABSTRACT

When MapReduce process mass-data,it highly abstracted parallel computing process on large clusters into two functions (Map and Reduce). so pre-organization input dataset and generating Map task view is a key step for processing.. In this paper, iterative reduction on the existing complex, large-scale task set based on rough set knowledge, get sub views equivalence class task after the update, calculate the optimal Properties based on the set with minimal time overhead, according to the optimal attribute set to delete redundant view, Finally the task combination view for parallel processing obtained after optimized,. Simulation results show that, compared with the reduction before optimization, MapReduce algorithm avoids unnecessary complexity in dealing with the same task, the running time and efficiency are better promotion, show the effectiveness of the method.

Keywords: Rough set, map-Task view, MapReduce, mass-data

INTRODUCTION

The increasing data in the data acquisition, storage and analysis has become a research hotspot. Some of the new storage system such as network database, cloud platform, extensible database to abandon the traditional management mode of relational database, and the data model is more simple and weak a characteristics to meet the large data in the expansion of demand. Where MapReduce is the common research and application is data processing method [1,2]. It is suitable for large scale analog to digital (more than 1TB) parallel computing. MpReduc will run on large scale complex parallel computing on clusters of highly abstract two function (Map and Reduce), each Map tasks and each Reduce task can run at the same time, a separate computing nodes, the processing efficiency is very high[3]. In MapReduce, the program prior to execution, to organize the input class will be scheduled in accordance with the Map the number of tasks the input file segmentation. The data set was divided, efficient mapping relationship between data and task is called the Map task view. How to obtain the Map task view well, many kinds of large data sets, pretreatment, is a hot spot to solve the practical application.

Pre processing method in recent years for large-scale data processing set mainly in the clustering algorithm for direct processing of the data itself, such as the method based on density and grid, combined with the bionic method, such as ant colony algorithm and clustering algorithm based on particle swarm algorithm, the center, such as K-means clustering[9]. The solution is to make the object class is as similar as possible, between the different object classes. This kind of algorithm canmake in dealing with large-scale growth structure data set effectively identifies the data the intrinsic structure. But the processing to the data set and the task set is linear discrete, easily trapped in local optima prematurely. In addition, also appeared the first task decomposition and combination of tools and methods of data collection, such as Aqualogic[2] and Damia[3], GPSO[4], this method can deal with large-scale cross domain task request, and has a relatively good flexibility and ease of use.However, they are either from the ideal state of single block grid, or in accordance with the characteristics of the data itself to organizing and repartition the original data set, to improve the efficiency of the algorithm and reduce the difficulty of.

When the data is large in scale, complex data structures, problems are: (1) the re organization of data and then split time cost greatly, also need to take extra storage space. (2) Such as fruit region is irregular mosaic, then the restructuring and re partition the excessive costs [5]. (3) On the polymerization and splits the branch data loss in the domain decomposition parallel program debugging, analysis and verification of information, is not conducive to the task.

In order to avoid these problems, this paper proposed a task view composition optimization method based on rough sets reduction. The method is based on rough set theory, applications of constructive method to approximate the spatial space derived from existing, design for task and data mapping of the Map view engine, hand to local fast update the view, one can view the description of association between pairs. The method is suitable for parallel processing program in most of the aggregation and domain decomposition method, especially can be dispersed in the communication merge multiple sub data set for a communication, in order to reduce the communication complexity.

PROBLEM ANALYSIS

2.1. *Knowledge reduction algorithm*

The rough set theory is a mathematical tool which can quantitatively analyze the imprecise, inconsistent and incomplete information and knowledge. Its basic idea is to form concepts and rules through classification and summarization of the relational database, then realize knowledge discovery through equivalent classification. Its most significant feature is that it did not need to provide any prior information besides the required processing data. Therefore, the uncertainty description or processing of the problem is quite objective. Currently, research based on Rough set theory mainly focus on attribute reduction rule acquiring and algorithm research. Attribute reduction, as an NP-Hard problem, has become a hot topic for many scholars. Reduction theory based on rough set developed rapidly over the past several years, many new and effective methods have come forth. For example, for different information systems (coordinated and uncoordinated, complete and incomplete), Pawlak, Wong, Yao and Iwinski have proposed many methods by combining information theory, concept lattice and swarm intelligence algorithm technology, such as data analysis method, attribute reduction algorithm based on information entropy, dynamic reduction algorithm, incremental algorithm and identified matrix algorithm. They all achieved corresponding results [5-9].

Below are some basic concepts of Rough set in this paper.

Definition1:Quintuple $S =< U,C,D,V,f >$ isa decision table,which $U = \{x_1, x_2,..., x_n \}$ represent the non-empty finite set of the objects, called the domain; subset C and D are called condition attribute set and decision attribute set; $C \cap D = \varnothing$, $V = \bigcup_{a=C\cup D} V_a$, V_a is the range of attribute a; $f : U \times (C \cup D) \to V$ is an information function, it specifies the attribute values of every object in U .

Definition 2: To $\forall a \in C \cup D, x \in U$, $f(x, a) \in \forall a$; each attribute subset $A \subseteq C \cup D$ determines a binary i indistinguishable relation: $IND(A) = \{(x,y) \notin U \times U \mid \forall a \in A, f(x,a) = f(y,a)\}$. Relation $IND(A)$ constitute a division of U , denoted as $U / IND(A)$, abbreviated U / A .each of the element $[x]_A = \{y \mid \forall a \in A, f(x,a) = f(y,a)\}$ is called equivalent class.

Definition 3: Assume U, V represent two domains. Elements u ∈U and v ∈V are compatible, denoted as $u \subset U$. Without loss of generality, it is assumed that for each u ∈U, there will be a v ∈V to ensure that they are associated, vice versa. Then the compatible relationship between U and V can be multi-value mapping, assign a value to each object, that is, to define it, i.e. $\forall(u) = \{v \in V \mid u \subset v\}$.

Definition 4: Set the decision table information system $S =< U,C,D,V,f >$, for each subset $X \subseteq U$ and uncertain relation A. the lower approximation sets and upper approximation sets of X can be defined by the basic set of A respectively as follows:

Lower approximation sets: $A_{-}(X) = \bigcup\{Y_i \in U \mid IND(A) : Y_i \subseteq X\}$

Upper approximation sets: $A^{-}(X) = \bigcup\{Y_i \in U \mid IND(A) : Y_i \cap X = \varnothing \}$

Definition 5: Assume C,D are attribute sets, no attribute of D can be omitted. If $D \subseteq C$, and $Ind(D) = Ind(C)$, then Q is a reduction of P , denoted as $Red(P)$. Furthermore, if $Core(C)$ is denoted as the attribute set that cannot be omitted, referred as the core of C .then all the reduction Red(C) just exactly equals the core of C, That is $Core(C) = \cap Red(C)$. The formula not only reflects that the relation between nuclear and all the reduction are obtained by reduction, it also shows that core is the most important part of knowledge base, which cannot be deleted in the process of knowledge reduction.

Definition 6: In decision table $S =< U,C,D,V,f >$, mark $U / C = \{[x_1^{'}]_c [x_2^{'}]_c ,...,[x_s^{'}]_c \}$.
$U^{'} = \{x_1^{'}, x_2^{'},..., x_s^{'}\}$ $U^{'}_{POS} = \{x_{i_1}^{'}, x_{i_2}^{'},..., x_{it}^{'}\}$, the object in $U^{'}_{POS}$ is compatible object, $U^{'}_{BND}$ equals $U^{'} - U^{'}_{POS}$, so $S^{'} = (U^{'} = U^{'}_{POS} \cup U^{'}_{BND}, C,D,V, f)$ is the simplified decision table.

Decision table can be divided into a consistent decision table and inconsistent decision table. When D is totally depend on ($C \Rightarrow D$), it is called consistent; when $C \Rightarrow kD(0 < k < 1)$, the decision table is inconsistent. Whether the decision table is reducible depending on whether it is a consistent decision table. This is because different reasons can cause the same results, but the same reason is not allowed to lead to different results.

2.2 *knowledge reduction algorithm based on Rough Set*

In the process of mass data, the core of the problem is the document segmentation and host resource selection algorithm. The document segmentation in logic only on the input data into pieces, and not on the disk will be cut into pieces for storage and processing. When the segmentation scheme is determined, a very important host resource selection. Common disposal method is in accordance with the price level, the task scheduling, a priority to the data processing of idle resources on the local node, if the node is not capable of processing data, the processing of the same domain data, the worst case is to deal with the other domain (but must be in the same data center).

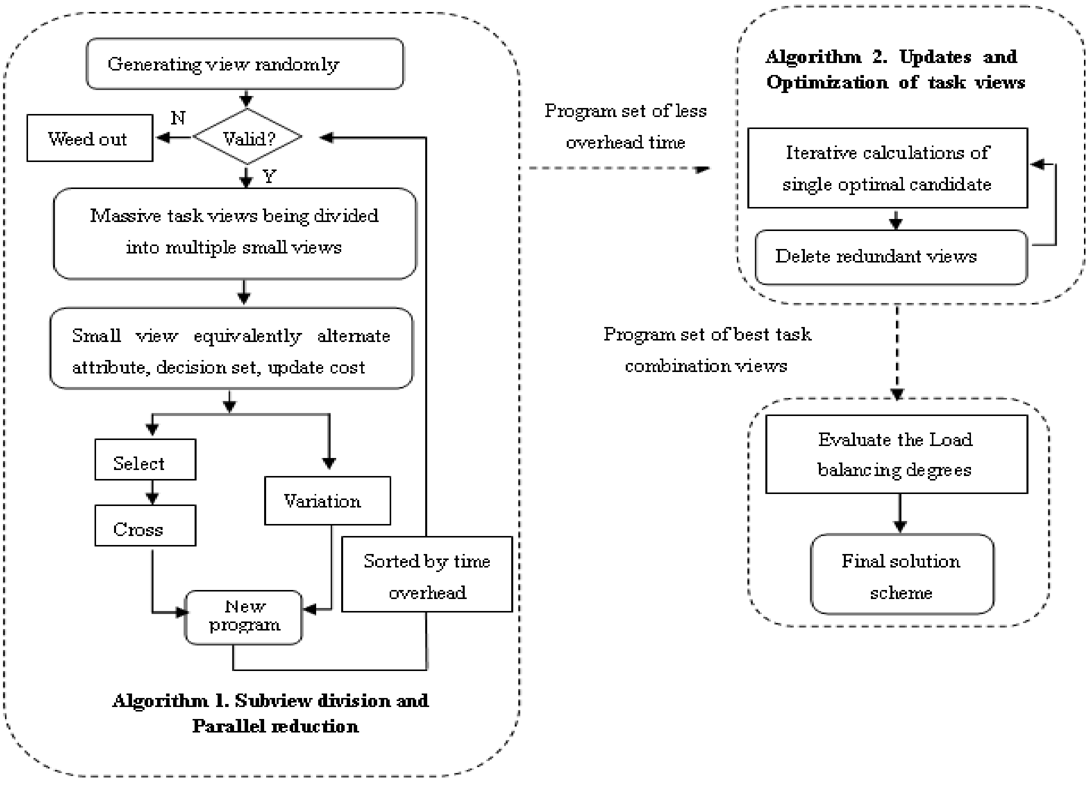
Generating view randomly
Valid?
N
Weed out
Y
Massive task views being divided into multiple small views
Small view equivalently alternate attribute, decision set, update cost
Select
Cross
Variation
New program
Sorted by time overhead
Algorithm 1. Subview division and Parallel reduction
Program set of less overhead time
Algorithm 2. Updates and Optimization of task views
Iterative calculations of single optimal candidate
Delete redundant views
Program set of best task combination views
Evaluate the Load balancing degrees
Final solution scheme

Figure 1. Algorithm flowchart

This problem can be in accordance with the 2.1, similar to the two parts:

Set for each data set F<file, start, length, and hosts>, blokcsize are four attributes, such as the occupied resource block numbers of block, goalsize as desired by the user file, Goalsize=totalsize/numsize, and minsize: is the minimum files can be divided.

F=max {minsize, min {goalsize, blocksize}}

For each resource set H<locality, block >, where locality local, in this paper is divided into the node locality, rack locality, data center locality three. In order to improve the local Map task view, should be the size equal to block and F, you can set set of conditions, including domain real-time parameters, such as server and physical distance and average network traffic, node locality, rack locality, data center locality related parameters.

A decision set, said request time sequence, the type of business, said the quality of service requirements, said the economic principles, said the source file size, said the task schedule length, said safety requirements.

A MAP-TASK VIEW GENERATION ALGORITHMS BASED ON ROUGH SET THEORY

The specific flow chart of map-task view generation algorithm shown in Figure1: Algorithm1 and Algorithm 2 in the figure1 is described in detail below.

3.1*Subview division and parallel reduction algorithm*

The traditional parallel reduction strategy assumes place all objects into the memory at one time. Yet this is not suitable for large-scale task view sets in Gloud storage system [10]. By using the MapReduce technology to handle massive amounts of data, we did not need to deal with fault tolerance processing and data partitioning. We just need to divide the actual problem into a number of parallel sub-problems. Its main functions involve Map function and Reduce function. Map function mainly deals with the calculation of different sub-equivalence class, while reduce function mainly calculates the number of unrecognized objects in the same equivalence class [10].

First, assume there are k different decision attribute values in decision table S , the decision attribute value of compatible objects respectively mapping $1,2,...,k$. That of incompatible objects all mapping $k+1$. In this way, the entire decision table S can be seen as constituted by $k + 1$ sub-decision table $D_1, D_2, ...D_k, D_{k+1}$.

Each decision table contains objects of the same category; the number of the objects is $n^1, n^2, ..., n^k$ respectively. Therefore, decision table S is "consistent" decision table.

Initialization: in the consistent decision table S, the recognizable objects in task combination views was generated by two objects with different decision attribute values and different condition attribute combination values. Assume $a \in C$, if the decision value of two objects is different, condition attribute a is also different, then a can identify these two objects, i.e. a recognizable object pair. In order to identify all the recognized objects in task scheduling views according to the above rules, for k different decision attribute values, mapped into $k+1$ sub decision table $T_1, T_2, ..., T_k, T_{k+1}$.

Following process is the reduction of one of the component.

Step 1: Calculate the condition mutual information of condition attribute C_i and decision attributes D_i in the decision table T_i

Step 2: Calculate the relative core $C_0 = Core_{Di}(C_i)$ of C_i relative to D_i . Generally, $I(C_0, D_i) < I(C_i, D_i)$; sometimes $C_0 = \varnothing$,then $I(C_0, D_i) < I(C_i, D_i) = 0$;

Step 3: Order $I(B_i, D_i) = I(C_i, D_i)$ repeat in conditions attribute set $C_i - B_i$.

□For each attribute $p \in C_i - B_i$, calculate the condition mutual information $I(p, D_i | B_i)$;

□ Choose the attribute that makes the condition mutual information $I(p, D_i | B_i)$ the biggest. Denoted as p (if the attribute are more than one, choose the one that has the least combination with attribute B_i); and $B_i - B_i \cup \{P\}$.

□If $I(B_i,\ D_i) = I(C_i,\ D_i)$,then terminate; otherwise turn to □;

Step 4: Finally B_i is a reduction of C_i relative to D_i

Following is an example of the knowledge reduction algorithm under the Gloud storage environment. Table 1 is a part of the typical decision table when the task view combines together, in which the condition attribute set $C=\{a_1,a_2,a_3,a_4,a_5,a_6,a_7\}$, a_1 indicates the order of requesting time, a_2 indicates the type of service, a_3 is the requirement of quality service, a_4 is the economic principle, a_5 is the size of the source file, a_6 is the length of task scheduling, a_7 is the security requirement.

Decision attribute set cc $D=\{d\}$ represents the preliminary results, domain $U = \{U_1,U_2,U_3,U_4,U_5,U_6,U_7\}$.

TABLE 1. Task view combines Decision table

task	a_1	a_2	a_3	a_4	a_5	$a_{6/ms}$	a_7
U_1	1	backup	3	maximize	high	4	Ordinary clients
U_2	2	Large Files division	2	maximize	low	2	VIP clients
U_3	3	backup	1	maximize	low	∞	Unexpected clients
U_4	4	download	1	null	medium	5	Trusted clients
U	5	Upload	3	null	high	600	Trusted clients
U_6	6	Search	3	null	Medium	1200	Unexpected clients
U_7	7	verify	2	null	High	100	New clients

Algorithm 1 is used in the attribute reduction of the object in table 1 with the minimum time overhead. First calculate $I(C,D)$=1.761, then calculate the core C relative to D, $C_0 = \{a_1\}$ $B=\{a_1,a_2,a_3\}$ will be obtained through step 3 by algorithm 1. Next judge the conditions $I(B,D) = I(C,D)$. If the condition is true, then algorithm end; and output the reduction set $B=\{a_1,a_2,a_3,a_5\}$ which is a set C relative to a set D.

Analyze the relative reduction set B .Because $H(D|\{a_1,a_2,a_3\}) = H(D|\{a_1,a_2,a_3,a_5\})$, so the attribute a_5 is the redundant attribute of reduction B relative to decision attribute set D. Thus, the reduced decision table can have less condition attribute while with no loss of knowledge content.

3.2 *Optimization algorithm of the task combination views.*

In order to optimize the task combination views, model it as a 0-1 programming problem. There are many ways to solve the problem quickly. Description of the 0-1 programming is as (1):

$$\min B_i \text{ s.t. } x^E \times ES' = I^S \qquad (1)$$

B_i is the target function; x^E is the combination programs chosen from the equivalent subview, which is 1 when chosen, otherwise 0. Constraints are an original task can only choose one corresponding equivalent set in the task combination program. I^S is a column vector whose length is |S| and elements are all 1. Since the target function does not meet the principle of superposition, iterative calculation of the single-view optimal attributes is selected. Some of the specific process of the algorithm will be introduced in another paper.

RESULTS AND DISCUSSION

4.1 *experimental platforms*

The proposed algorithm in this paper was conducted on the school distributed storage laboratory witch built by the open-source platform Hadoop 0.20.2 and Java 1.6.0_20. We deployed a self-developed heterogeneous local storage system, the directory which ipnuted has six files file1, File2, file3, file4, file5, file6 defaulty, one of the experiments are count word frequency for each word appears in a document, Another experiment is comparing the performance of different algorithms when they download the same large video file. The initialization parameters as shown in table 2.It support uses including 108 clients, 3 servers for job scheduling tasks. The parameters of task view equivalent class at some point is shown in Table 3 .

Table 2. Task view equivalence class at some point view combines Decision

name	minsize	goalSize	splitSize	File Corresponding number F
file1	1MB	totalSize	64 MB	4
File2	4MB	totalSize/5	50 MB	5
File3	128MB	totalSize/2	128 MB	2
File4	256 MB	totalSize/5	64 MB	6
File5	1G	totalSize/8	128 MB	5
File6	4G	totalSize/4	256 MB	16

Table 3. Task view equivalence class at some point view combines Decision

Node ID	Number of tasks	Number of condition attributes	Set value of the number of decision attributes number
0	8249	10000	103
02	680	9	2
68	785	26	5
103	430	30	10
3	799	78	5
79	101	11	3

4.2 *Experimental results*

Traditional data processing algorithm in the application of data processing intensive is inadequate, including poor scalability, flexibility and performance is poor. In recent years there has been a massive data processing methods such as MapReduce, for parallel computation of large data sets. In this paper, the application of rough knowledge reduction focused on the existing complex, large-scale data sets of task mapping iterative reduction, get sub view equivalence class task after the update, time cost of task sets the optimal attributes were calculated based on, according to the optimal attribute set to delete redundant view, finally get the map like task combination view after optimization, for parallel processing. Simulation results show that, compared with the reduction before optimization, MapReduce algorithm avoids unnecessary complexity in dealing with the same task, figure 4 to figure 8, the time span from the time cost, operation time, speedup, scalability to measure the same computing tasks combined views about Jane after effect. As you can see from Figure 4, the tasks view are reduced after optimization than before optimization, was split on the file before optimization, the communication overhead is almost 10 times after optimization, this fact reveals the characteristics of the low overhead algorithm. Furthermore, its running time as the number of attributes has been an obvious increase rising trend; and with the task set size is fixed, and the file number increased, the algorithm to the optimization of good speedup. When a node and task set size while expanding, expandable algorithm is also very good. Therefore, the rough set algorithm for task combination view attribute reduction can meet the need of large-scale storage system based on the proposed, has a good application prospect.

We mainly measured the effects of reduced task combination views under the environment of Gloud computing from time span, runtime, speedup ratio and scalability. Figures 4 to 7 is the comparison before and after optimization. As we can see from the figure, runtime increased rapidly as the number of attributes goes up. When the scale of task set is fixed, this algorithm has better speedup ratio as the number of node increases. When the size of task set scale and number of node increase at the same time, the scalability of the algorithm is also very good. Therefore, the proposed task combination view reduction algorithm based on rough set knowledge is capable of applying in large-scale storage systems and has a better application prospects.

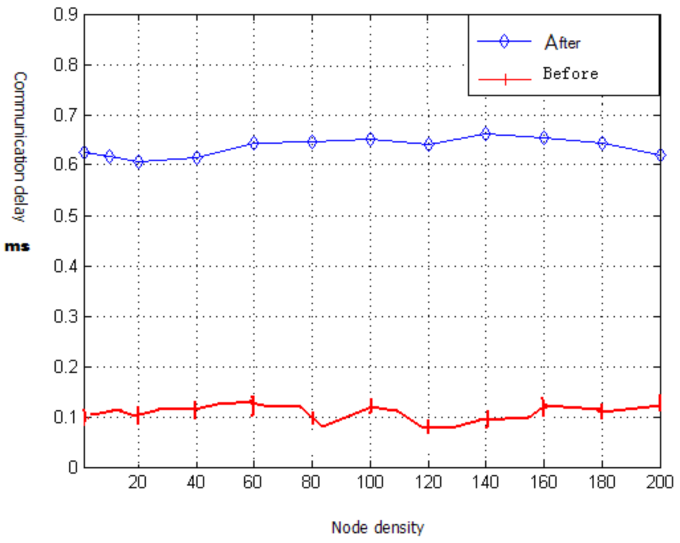

Figure 4. File segmentation time expenditure Comparison

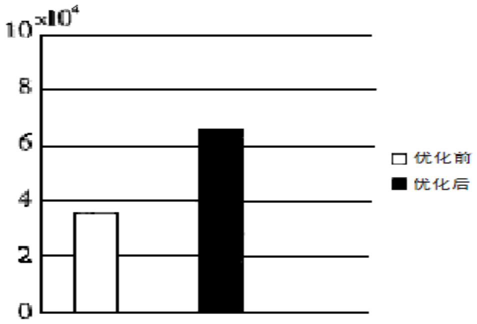

Figure 5. Comparison in time span

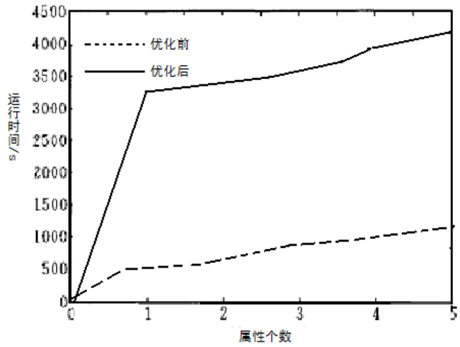

Figure 6. Comparison in runtime

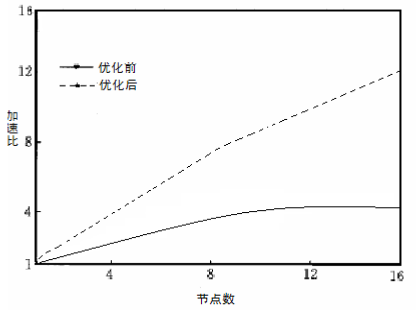

Figure 7. Comparison in speedup ratio

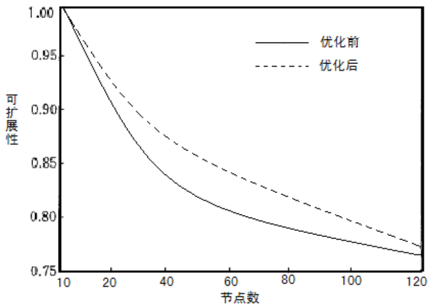

Figure 8. Comparison in scalability

Acknowledgements
This work was supported by the Science Research Foundation of Guangxi University of Scienceand Technology (No.1261126), Guangxi Natural Science Foundation (No.2013GXNSFBA019268, No.2011GXNSFA018162) ， Guangxi features professional and curriculum integration funded construction projects (No.GXTSZY217).

REFERENCES

[1] Gelan Yang, Huixia Jin, and Na Bai.*Mathematical Problems in Engineering*, 2013, vol. **2013**, Article ID 272567, doi:10.1155/2013/272567.
[2] Tang Peihe, Xuyiyi.*Journal of Convergence Information Technology*. **2012**, 7(16):393-400.
[3] Chen L. *Applied Mechanics and Materials*, **2014**, 443: 599-602.

[4] Destercke S. *Soft Computing*. **2012**, 16(5): 833-844.
[5] Gao Liqun, Li Ruoping, Zou spin. *Northeastern University (Natural Science)*. **2011**; 32(11):1538-541.
[6] Wu, Yue, Gelan Yang, Huixia Jin, and Joseph P. Noonan. *Journal of Electronic Imaging*. **2012**, 21(1): 013014-1.
[7] Gelan Yang, Huixia Jin, and Na Bai. *Mathematical Problems in Engineering,* **2014**, vol. 2014, Article ID 632060, 13 pages, doi:10.1155/2014/632060.
[8] Xiao Fu, Jin Liu, Haopeng Wang, Bin Zhang, Rui GAO.*Lecture Notes in Computer Science*. **2011**, 76(66):123-131.
[9] Stavrinides G L, Karatza H D. *Simulation Modelling Practice and Theory*, **2011**, 19(1): 540-552.
[10]YANGGelan, Yue WU, Huixia JIN. *Journal of Computational Information Systems*, **2012**, 8(10): 4315-4322.
[11]Xu Y. *TELKOMNIKA Indonesian Journal of Electrical Engineering*, **2014**, 12(3): 2088-2095. *Science and Technology (Natural Science).* 2005, 33(8):30-33.