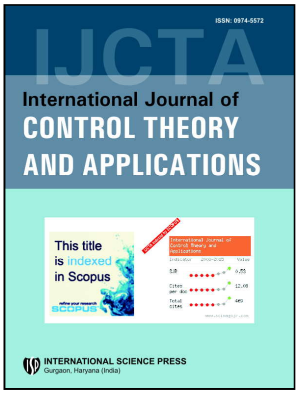

International Journal of Control Theory and Applications

ISSN : 0974-5572

Volume 10 • Number 18 • 2017

Availability and Reliability Analysis of Computer Systems

Jitendra Kumar[1], Vikas Shinde[1] and Mukta Kalra[2]

[1] Department of Applied Mathematics, Madhav Institute of Technology & Sciences, Gwalior (M.P.) INDIA -474005, Emails: jkmuthele@gmail.com, v_p_shinde@rediffmail.com
[2] Department of Mathematics, Govt. Narmada P. G. College, Hoshangabad (M.P.) INDIA – 461001, Email: kalramukta@yahoo.com

Abstract: System availability and reliability is a major concern in computer systems design and analysis. We observed the availability analysis for computer system with various issues. Markov model is developed and equations are derived to obtain the steady state availability. Both software and hardware failure are considered. Source to destination network analysis is derived. SYREL algorithm is used to obtain compact terminal reliability expressions between a terminal pair of computers of complex networks.

Keywords: Availability, Reliability, Cutset.

1. INTRODUCTION

Recent development of computer technology and appearance of new advance communication applications, assured quality of service, network availability and reliability has become a critical issue. In order to achieve desired grade of availability and reliability of computer networks such as restoration, mitigation and prevention to realize fault tolerance during network operation. Although the techniques make a great number of repetitions to operate the capacity of network a new generation in running process also increase the system complexity. This increased complexity of computer network raises the occurrence diversity and severity of failures in important applications. Several authors have discussed these issues. A general model for centralized heterogeneous distributed system and the distributed service reliability which is defined as the probability of successfully providing the service in a distributed environment an important performance measure for this type of systems is investigated by Dai et al. [1]. Dirk et al. [2] considered network reliability optimization with network planning, where the objective is to maximize the network's reliability, subject to fixed budget. Distefano and Puliafito [3] analyzed dynamic system reliability and availability with DRBDs. Gruber and Keane [4] proposed an application of genetic algorithms to the output from reliability availability & maintainability models of complex (mainly industrial) systems to reduce computational costs while maintaining a desired level of fidelity of the model. Hariri and Raghvendra [5] gave a symbolic reliability algorithm based on path and cutest methods. Huang & Hung [6] applied queueing models to describe the fault detection and correction processes during software development and proposed an extended infinite server queueing model with multiple change points to predict and evaluate software reliability. Jain and Singh [7] gave the reliability of repairable multi component redundant system. Joseph et al [8] presented the development of an integrated methodology for quality and reliability improvement when degradation data are available as the response in the experiments. Jian and Shaoping [9] developed generalized stochastic Petri net (GSPN) model for availability of nodes in the networks through path selection and service performance. Lai et al [10] described system availability Markov model for a simple two host system and a general multi-host system in distributed computing environment. Raghavendra et al [11] presented a reliability of a distributed processing system in an important design parameter that can be described in terms of the reliability of processing elements and communication links and also of the redundancy of programs and data files. Reijns et al [12] discussed a low-cost, compositional approach based on the use of the first four statistical moments to characterize the failure time distributions of the constituent components, subsystems and top-level system. The approach is based on the use of Pearson Distributions as an intermediate resolving vehicle, in terms of the constituent failure time distributions are approximated. Shinde et al [13] considered the three different series system configurations with mixed standby (including cold and warm) components.

The remaining of the paper is as follows. In section II, Availability Analysis of the model is described by stating requisite notations. Source to destination network availability is derived in section III. In section IV, we discuss CDP algorithm procedure and SYREL for evaluating the reliability analysis of complex computer systems. Finally conclusions are drawn in section V.

2. AVAILABILITY ANALYSIS

2.1. Notations

A_i : Component Availability model,

MTBF : Average /Mean time between successive component failures,

MTTR : Average/Mean time of repairs,

A(t) : Probability that the component is in a state of physically correct operations and do not have any software problems,

R(t) : Reliability

X_i : Indicator variable for component i,

$P_j(t)$: Probability that at time t the component is in state j,

$P'_j(t)$: Derivative of $P_j(t)$ with respect to t,

λ_h : Hardware failure rate of X_i from the healthy state (constant),

λ_s : Software failure rate of X_i from the healthy state (constant),

λ_{sh} : Hardware failure rate of X_i from the software failure rate (constant),

λ_{hs} : Software failure rate of X_i from the hardware failure rate (constant),

μ_h : Hardware failure repair rate of X_i (constant),

μ_s : Software failure repair rate of X_i (constant),

μ_{sh} : Hardware failure repair rate of X_i from the software failure repair rate (constant).

Availability is defined as the probability that a system can accept and respond in an expected manner to requests for computing services. The metric will be derived by first developing availability models for the components in the system. In this two level hierarchical structure the results of the first level component models become the inputs for the second level environment model. In component models processors are considered from the point of view of hardware and operating system (software) failures. Storage devices are modeled only for hardware – functions. Networking model deal with hardware/software failure and congestion.

In general the measure of component availability depends not only on hardware failures but also on the failure rates due to software problems.

Set of following differential equations for the component can be obtained from the state diagram.

$$P_0'(t) = -(\lambda_h + \lambda_s)P_0(t) + \mu_h P_1(t) + \mu_s P_2(t) \quad (1)$$

$$P_1'(t) = \lambda_h P_0(t) - (\mu_h + \mu_{sh})P_1(t) + \lambda_{sh} P_2(t) \quad (2)$$

$$P_2'(t) = \lambda_s P_0(t) + \mu_{sh} P_1(t) - (\mu_s + \lambda_{sh})P_2(t) - \lambda_{hs} P_3(t) \quad (3)$$

$$P_3'(t) = \lambda_{hs} P_2(t) \quad (4)$$

It is evident from the above equations that the rate of probability of staying in any state is equal to the sum of frequencies of input into that state minus the sum of frequencies of output from that state.

At t = 0, $P_o(0) = 1$, $P_1(0)$, $P_2(0)$, $P_3(0) = 0$

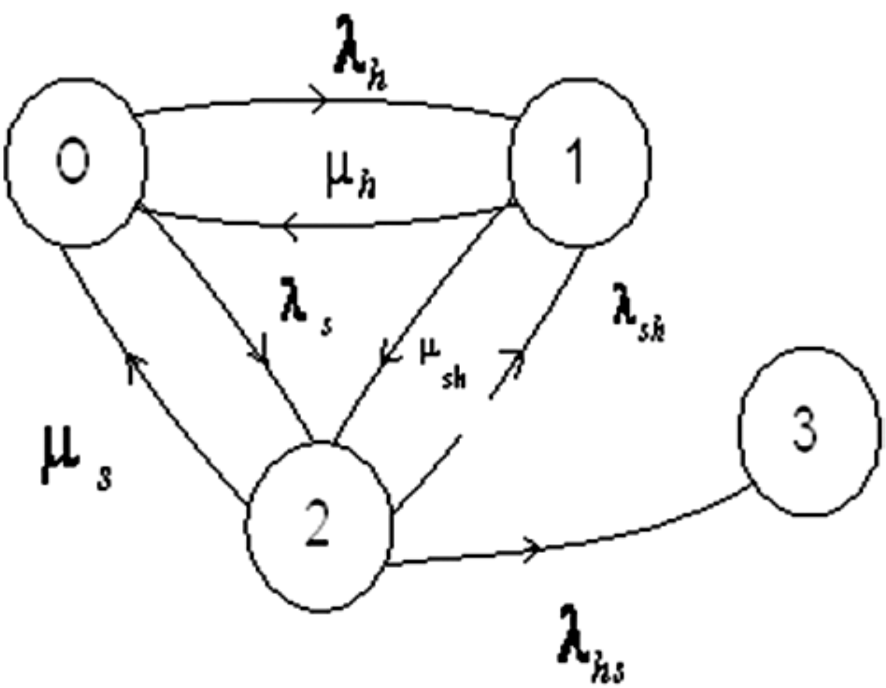

Figure 1:

As we know that the time dependent component availability A(t), is the probability that the component will be in state 0. i.e.

$$A(t) = P_0(t) \quad (5)$$

The probability that a continuous-time Markov process will be in state j at time t is evaluated by a limiting value which is independent as a initial state.

$$A = \lim_{t \to \infty} A(t)$$

for steady state condition taking t tends to ∞ in equation (1), (2), (3) and (4) equations reduces in

$$-(\lambda_h + \lambda_s)P_0 + \mu_h P_1 + \mu_s P_2 = 0 \tag{6}$$

$$\lambda_h P_0 - (\mu_h + \mu_{sh})P_1 + \lambda_{sh} P_2 = 0 \tag{7}$$

$$\lambda_s P_0 + \mu_{sh} P_1 - (\mu_s + \lambda_{sh})P_2 - \lambda_{hs} P_3 = 0 \tag{8}$$

$$\lambda_{hs} P_2 = 0 \tag{9}$$

On the basis of such considerations, the algorithm uses a different color image multiplied by the weighting coefficients of different ways to solve the visual distortion, and by embedding the watermark, wavelet coefficients of many ways, enhance the robustness of the watermark.

Solving these equations, we get the following component availability metric:

$$\text{A} = P_0 = \frac{\lambda_{hs}(\mu_h + \mu_{sh})}{\mu_{sh}(\lambda_{hs} + \lambda_s + \lambda_h) + \lambda_{hs}(\lambda_h + \mu_h) + \mu_h \lambda_s} \tag{10}$$

3. SOURCE -TO- DESTINATION AVAILABILITY MODEL

Network availability is derived on the node and trunk topology. When a packet enters the networks it is stored at its source node. The packet is then routed to an outgoing trunk according to a routing procedure. When the outgoing trunk is free the packet is sent to the next node in its path. This process is repeated whenever the packet reaches its destination node. For a given source to destination network availability is derived based on the min path set and min cut set method. Algorithm starts by identifying the all possible paths between any pair of nodes and then deriving a Boolean expression for the availability metric calculation. This expression is then employed to obtain a source to destination availability measures by utilize the corresponding individual component availabilities by assuming the dynamic routing in computer networks, shown in fig. 2.

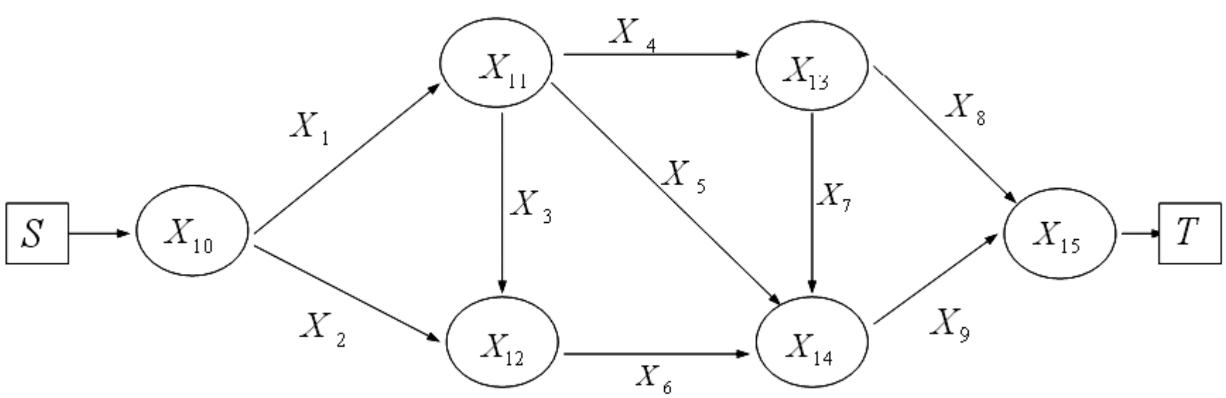

Figure 2:

In fig 2 five different paths exists between the source host S and destination host T.

Path 1: $X_{10}X_1X_{11}X_4X_{13}X_8X_{15}$

Path 2: $X_{10}X_1\ X_{11}X_5X_{14}X_9X_{15}$

Path 3: $X_{10}X_2X_{12}X_6\ X_{14}X_9\ X_{15}$

Path 4: $X_{10}X_1\ X_{11}X_3\ X_{12}X_6\ X_{14}X_9\ X_{15}$

Path 5: $X_{10}X_1\ X_{11}X_4\ X_{13}X_7\ X_{14}X_9\ X_{15}$

Min Path Set is

$X_{10}X_1X_{11}X_4X_{13}X_8X_{15}+X_{10}X_1X_{11}X_5X_{14}X_9X_{15}+X_{10}X_2X_{12}X_6X_{14}X_9X_{15}+$

$X_{10}X_1\ X_{11}X_3X_{12}X_6X_{14}X_9\ X_{15}+\ X_{10}X_1\ X_{11}X_4X_{13}X_7X_{14}X_9X_{15}$

The end-to-end availability between S and T is represented by the probabilistic Boolean expression

P{ path 1 up}+

P{path 1 down}P{path 2 up}+

P{path 1 & 2 down}P{path 3 up}+

P{path 1 & 2 & 3 down}P{path 4 up}+

P{path 1 & 2 & 3 & 4down}P{path 5 up}

Assuming that individual component failure and repair rates are statistically independent from each other, the availability measure of the preceding expression is as follows

$A(S\rightarrow T) = A_1A_4A_8A_{10}A_{11}A_{13}A_{15}+$

$(1-A_4A_8A_{13})\ A_1A_5A_9A_{10}A_{11}A_{14}A_{15}+$

$(1-\ A_1A_4A_8A_{11}A_{13})(1-\ A_1A_5A_{11})A_2A_6A_9A_{10}A_{12}A_{14}A_{15}+$

$(1-\ A_4A_8A_{13})(1-\ A_5)(1-\ A_2)A_1A_3\ A_6A_9A_{10}A_{11}\ A_{12}A_{14}A_{15}+$

$(1-\ A_5)(1-\ A_8)(1-\ A_2A_6A_{12})(1-\ A_3A_6A_{12})\ A_1A_4A_9A_{10}A_{11}\ A_{13}A_{14}A_{15}$

6. RELIABILITY ANALYSIS

Terminal reliability is an important parameter in the design of highly reliable computer networks. Terminal reliability is a probabilistic measure and is defined as the probability that there exists one operative path between a given pair of computers. Determining symbolic expression for the terminal reliability is important. There are two reasons (i) several computer networks have fixed topology, however the reliability of its elements changes with time. Revaluation is easy with symbolic expression and the sensitivity to changes in element reliabilities can be readily determined. (ii) Some applications are desired to improve the terminal reliability of a network under a given cost of constraint.

6.1. Cutset Disjoint Procedure (CDP)

Cutset is a set of edges which disconnects the connection between the input and output of the network, it is minimal if it does not contain a subset which itself is a cutset of the network. Network reliability is the disjoint sum of probability of success of all the minimal paths leading from the source node to terminal node. The probability of success of a network suggests that for evaluation of network reliability, the enumeration of all the minimal paths should be the first essential step then disjoint sum of all such minimal paths should give the probability of success. A complementary statement can be made for unreliability evaluation in a similar manner. Therefore, for unreliability evaluation the first essential step would be the enumeration of all minimal cut sets then obtain the disjoint sum of all such cut sets to get the unreliability or the probability of failure of a system. For enumeration of minimal cut sets there are some methods one is based on graph theory and other one is inter conversion and minimization technique. Here we have used second method for finding minimal cut sets.

6.1.1. CDP Algorithm

Initialization $DC_1=C_1$;

for j = 2 to nc do; nc is the number of cut sets.

begin

$DC_j = C_j$

for i = 1 to j-1 do

begin

for for all dc_k ε DC, do

if dc_k is disjoint with C_i, then

select next dc_{k+1} ε DCj.

else

begin; make dc_k disjoint with C_i,.

$C_{ilk} = C_i - dc_k$

if $C_{i|k} = \Phi$ then

drop dc_k from list CD_j

else

replace dc_k with

$$\{\bar{e}_1, dc_k\}, \{e_1, \bar{e}_2, dc_k\}, \ldots\ldots\ldots\{e_1, e_2, \ldots\ldots e_{njk-1}, e_{njk}, dc_k\}$$

add them to the list DC_j.

end

end

add DCj to the list DC

end.

6.2. Symbolic Reliability Algorithm (SYREL)

This algorithm is used to determine the compact reliability expressions in complex computer networks. This algorithm incorporates conditional probability; set theory and Boolean algebra in a distinct approach in which most of the computational performed are directly executable Boolean operations. The conditional probability is used to avoid applying at each iteration the most time consuming step in reliability algorithms. So, SYREL Algorithm is applied the network which is derived in fig. 2.

6.2.1. SYREL Algorithm

Step1. Enumerate all paths between terminal nodes and sort them according to their lengths.

Step2. Initialization step.

$T_1 = Pr(E_1)$; $R = T_1$; i = 1

Step3. Updating Step.

i = i+1

Step4. Test path length (pl)

if pl_i = n- 1 then

$$T_i = \prod_{for\ all\ x_{k,l} \in S_i} (p_{k,l}) \prod_{for\ all\ x_{k,l} \notin S_i} (q_{k,l});$$

go to step 8

Step5. Determining minimal conditional sets.

for j=1 to i-1 do

$S_{j|i}$ = Sj - Si; add $S_{j|i}$ to list Li

for k = 1 to i - 2 do

begin * eliminate all absorbable conditional sets.

for j = k + 1 to i - 1 do

temp = $S_{k|i} \cap S_{1|i}$

if temp = $S_{k|i}$, delete $S_{j|i}$ from list Li

if temp = $S_{j|i}$, delete $S_{k|i}$ from list L,

end

Step6. If for all $S_{k|i}$, $S_{l|i}$ ε Li $S_{k|i} \cap S_{l|i} = \Phi$ then

$$Ti = Pr (Ei) . \prod_{for\ all\ S_{k|i} \in M_i} Pr(E_{k|i}); \text{ go to step 8}$$

Step7. If for all S_{kli}, S_{lli} ε Li $S_{kli} \cap S_{l|i} \neq \Phi$ then

Step8. Evaluate cutsets and then call CDP

$$Ti=Pr (Ei) \left[\sum_{for\ \ all\ DC_k} Pr(\overline{DC_k}) \right].$$

Step9. Evaluating terminal reliability.

R =R + T_i,

Step10. If i < m go to step 3

Step11. Stop.

6.2.2. An Illustrative Example

The set of paths between the terminal nodes (X_{10} - X_{15}) of fig. 2 and their corresponding sets are

P_1 : X_1 X_4 X_8

P_2 : X_1 X_5 X_9

$P_3 : X_2\ X_6\ X_9$

$P_4 : X_1\ X_3\ X_6\ X_9$

$P_5 : X_1\ X_4\ X_7\ X_9$

$S_1 = \{ X_1 . X_4 . X_8 \}$

$S_2 = \{X_1 .X_5 . X_9 \}$

$S_3 = \{ X_2 . X_6 . X_9 \}$

$S_4 = \{ X_1 . X_3 . X_6 . X_9 \}$

$S_5 = \{X_1 . X_4 . X_7 .X_9 \}$

Terms corresponding to each iteration of SYREL for the network shown in fig. 2 are as follows:

$i = 1; T_1 = p_1 p_4 p_8$

$i = 2;\ S_{1|2} = \{X_4, X_8\}$

$T_2 = p_1 p_5 p_9 (1 - p_4 p_8)$

$i = 3;\ S_{1|3} = \{X_1, X_4, X_8\}\ S_{2|3} = \{ X_1, X_5\}$

$DC_1 = \{ X_1, X_2\}$,

$DC_2 = \left\{X_1, X_6, \overline{X_2}\right\}$,

$DC_3 = \left\{X_1, X_9, \overline{X_4}, \overline{X_6}\right\}$,

$DC_4 = \left\{X_4, X_5, X_6, \overline{X_1}, \overline{X_2}, \overline{X_9}\right\}$

$DC_5 = \left\{X_2, X_3, X_4, X_5, \overline{X_1}, \overline{X_6}, \overline{X_9}\right\}$

$DC_6 = \left\{X_5, X_6, X_7, X_8, \overline{X_1}, \overline{X_2}, \overline{X_3}, \overline{X_4}, \overline{X_9}\right\}$

$DC_7 = \left\{X_2, X_3, X_5, X_7, X_8, \overline{X_1}, \overline{X_4}, \overline{X_6}, \overline{X_9}\right\}$

$$T_3 = p_2 p_6 p_9 \{p_1 p_2 + p_1 p_6 q_2 + p_1 p_9 q_4 q_6 + p_4 p_5 p_6 q_1 q_2 q_9 + p_2 p_3 p_4 p_5 q_1 q_6 q_9 + p_5 p_6 p_7 p_8 q_1 q_2 q_3 q_4 q_9 + p_2 p_3 p_5 p_7 p_8 q_1 q_4 q 6 q_9\}$$

$$i = 4;\ S_{1|4} = \{X_4, X_8\},\ S_{2|4} = \{X_5\},\ S_{3|4} = \{X_2\}$$

$$T_4 = p_1 p_3 p_6 p_9 q_5 q_2 (1 - p_4 p_8)$$

$$i = 5;\ S_{1|5} = \{X_8\},\ S_{2|5} = \{X_5\},\ S_{3|5} = \{X_2, X_6\}, S_{4|5} = \{X_3, X_6\}$$

$$T_5 = p_1 p_4 p_7 p_9 q_5 q_8 (1 - p_2 p_6)(1 - p_3 p_6)$$

$$R = \sum_{i=1}^{5} T_i$$

7. CONCLUSION

In this paper, we determined the availability and reliability analysis for computer system. The Markov model is developed for the availability analysis of computer system. We also obtained network analysis by using Boolean algebra. Reliability algorithm (SYREL) is used to make a set of paths or cutsets mutually exclusive. The critical step is not executed at each iteration of the algorithm and it is performed on a small number of sets. This provides a significant improvement in the execution time of SYREL when compared to the algorithms that apply directly the critical step to the set of paths or cutsets at each iteration because of implementing conditional probability to reduce the number of variables that should be considered during the decomposition process. Such model can be used to reduce node and trunk redundancies in the network and identify hardware failures from software failures. This may give so many advantages over distributed systems, including high throughput cost or performance benefits and potential for enhanced reliability of the network.

REFERENCES

[1] Y. S. Dai, M. Xie, M., K.L. Poh and G. Q. Liu, A study of service reliability and Availability for distributed systems, Journal of Reliability Engineering & System Safety, 79(1), 103-112, (2003).

[2] P. K. Dirk, Kin-Ping Hui, and N. Sho, Network Reliability Optimization via the Cross-Entropy Method, IEEE Transactions on Reliability, 56(2), 275-287, (2007).

[3] S. Distefano, and A. Puliafito, Reliability and availability analysis of dependent dynamic systems with DRBDs, Reliability Engineering and System Safety, 94(9), 1381-1393, (2009).

[4] A. Gruber and A. J. Keane, Optimization of system resources in reliability availability & maintainability problems using genetic algorithms. In, *16th International MIRCE Symposium,* Exeter, UK, 06 - 08 Dec 2006, UK, 14, 1-14, (2006).

[5] S. Hariri and C. S. Raghvendra, A Symbolic Reliability Algorithm Based on Path and Cutset Method, IEEE Transactions on Computers, 36(10), 1224-1232, (1987).

[6] C. Huang and T. Hung, Software reliability analysis and assessment using queueing model with multiple change points, Computers and Mathematics with Application, 60(7), 2015-2030, (2010).

[7] M. Jain and C. J. Singh, Reliability of repairable multi component redundant system, International Journal Engineering, 3, 107-114, (2000).

[8] V. R. Joseph, and Y. I-Tang, Reliability Improvement Experiments with Degradation Data, IEEE Transactions On Reliability, 55(1), 149-157, (2006).

[9] S. Jian and W. Shaoping, Integrated availability model based on performance of computer networks, Journal of Reliability Engineering & System Safety, 92 (1), 341-350, (2007).

[10] C. D. Lai, M. Xie, K.L. Poh, Y. S. Dai and P. Yang, A model for availability analysis of distributed software/hardware systems, Information and Software Technology, 44, 343-350, (2008).

[11] C. S. Raghvendra and S. V. Makam, Reliability modeling and analysis of computer networks, IEEE Transactions on Reliability, 35(2), 156-160, (1986).

[12] G. L. Reijns, J. C. Arjan and V. Gemund, Reliability Analysis of Hierarchical Systems Using Statistical Moments, IEEE Transactions on Reliability, 56(3), 525-533, (2007).

[13] V. Shinde, G. C. Sharma, and M. Jain, Prediction of availability for repairable systems with mixed standby components, Indian Journal of Mathematics and Mathematical Sciences, 2(2), 187-200, (2006).