ISSN : 0974 - 7435 Volume 10 Issue 15

2014

BioTechnology
An Indian Journal

FULL PAPER
BTAIJ, 10(15), 2014 [8822-8830]

RC algorithm in multimedia software transmission

Zhang Hua*, Li Yuefei
Hunan Information College, Changsha, Hunan, 410151, (CHINA)

ABSTRACT

Aiming at the problems in service oriented real-time multimedia software transmission, such as a long delay, frequent jitter and low reliability, this paper proposes RC algorithm RC estimates the dynamic data volume on service nodes and links through reasonable data volume approximation. And the data splitting technique is imported to convert the node delay into the edge delay. In the mean time, the reliability of nodes is considered so that the multimedia delivery problem is transformed into a conventional shortest path problem. The experimental simulation results show that the algorithm is a kind of high efficient and reliable algorithm spending smaller complexity achieving good path selection performance.

KEYWORDS

Multimedia; Approximation of the amount of data; Node; Price label.

INTRODUCTION

Real-time multimedia transmission system is widely used in real-time monitoring system, video conferencing and the others[1]. However, with the development of the mobile devices' ability and wireless communication technology, the traditional single network environment gradually becomes heterogeneous equipment and pervasive computing environment made up of network. Multimedia transmission has obvious differences with traditional file transmission, the transmission of traditional file for transmission delay, jitter is not too many requirements, but there are strict error control and retransmission mechanism. In the transmission of real-time multimedia transmission, demanding synchronicity, and requires small transmission delay. Multimedia transmission can endure packet loss caused by the error and abnormal, can endure due to no retransmission or error correction mechanism of packet loss or latency, but it will not tolerate caused by error control mechanism based on retransmission display discontinuity or confusion. In the pervasive environment, the difference of different links' transmission capacity is huge, and in particular, wireless link bandwidth is smaller and unstable; the difference of each node's processing capacity is huge, and in particular, the mobile node's processing power is often weak[2-4]. These features cannot ensure the reliability of real-time multimedia transmission system.

Multimedia network transmission technology, however, because of the multimedia compression, compression and transmission problem not solve, have not been able to achieve the ideal effect. In recent years, multimedia communication technology

SOA uses the available service to buiRC loose coupling application. As the key technology of SOA, service composition can combine independently distributed and available basic service to meet user's complex business requirements, which makes it adapt to the pervasive computing environment[5-8]. These characteristics help SOA use the existing distributed service resources to dynamically buiRC loose coupling multimedia transmission system in pervasive environment.

In the service-oriented real-time multimedia transmission system, a number of multimedia services deploy in different nodes in the network beforehand. These services can be divided into functional and non-functional services. Features services meet the functional requirements of users through subtitle's embedment, tags' addition and codes' conversion and the other necessary processing for the source information[9-11]. Non-functional services can reduce the amount of data transmission and reduce the time delay of data transmission through media compression and sampling, etc. In particular, for mobile users using wireless access, these non-functional services can effectively improve the achieved services' reliability of users, but it has the function itself to deal with time delay and reduce the multimedia video's quality. The function of the system is as follows. When receiving the users' request, the system starts from the data source and buiRCs a multimedia information processing chain after a number of functional and nonfunctional service processing nodes so as to transfer multimedia information with relatively low latency and high reliability to end users[12]. In this paper, it is assumed that the user's service requests arrive one by one and they don't affect each other.

In the service-oriented real-time multimedia transmission system, the service requests having the same functional requirements is often able to complete by a variety of service composition ways. At running time, each service component can choose different service counterparts to achieve. In real-time multimedia system, the users are very sensitive to time delay so the user cannot accept long delay and frequent jitter of multimedia information. In addition, service users can't control service because services are also affected by equipment, environment and other factors in the pervasive environment. The services have great differences between individuals and the reliability of the individual services will eventually determine the expected performance of composite services. Therefore, it is a challenging problem about how to choose a series of appropriate service copies to constitute high credible service path with the low latency in pervasive environment.

Zeng and the others think the service copies shouRC be chosen at running time rather than designed moment, because a lot of basic service's quality parameters are unavailable before running, and then he transforms the problem into linear optimization problem driven by quality[13]. Gu and the others propose an integrated P2P service composition framework---SpiderNet, which chooses service copies hop by hop through heuristic methods so as to satisfy the multiple quality requirements[14]. Besides, SpiderNet also guarantees the load balancing in certain degree. Considering the load balancing among service copies, Roman and the others proposes LIAC algorithm (further - inverse - the available capacity) so as to guarantee the load balancing among service copies in the choice of paths and the piggybacking mechanism is used to make LIAC more efficient [15-17]**.** Roman and others also puts forward the service portfolio fault-tolerant system based on service network in the WAN. Kalasapur and the others propose a dynamic service composition framework using hierarchical service network to handle mobility and dynamic in pervasive environments[18].

Based on real-time multimedia transmission system, this paper inspects the service path's selection criteria of reliable multimedia transmission from the two angles. The first is the real time. The users want to get the real-time data, but if there is a lot of data transmission delay, it won't be reliable to the users. The second is reliability. Due to the dynamic and heterogeneous of environment, each service copy may have errors and if the service paths chosen by system go wrong, the system has to choose a service path again to complete the service request, which can increase system loading. Besides, for the users, path switching also increases time delay further, so the service provided by the system is no longer reliable. It is the aim of this paper that when the service requests are given, the appropriate service copies are selected to buiRC multimedia transmission path with low latency and high reliability.

MULTIMEDIA TRANSMISSION MODEL

Problem Definition

Each service replica can be expressed as a four element $\left(f, r, o, e\right)$, in turn, said compression type service to the service rate, copy unit processing time and reliability. For example, s_1^2 is expressed as $\left(f_1, r_1, o_{1,2}, e_{1,2}\right)$. The goal of the system is in the choice of a time delay from s_0 to s_t in all paths as small as possible, and the reliability is as large as possible path. In only considering the delay, due to changes in the amount of data, which is equivalent to a multi constrained path (multi-constrained path, referred to as MCP) problem. Wang proved that this problem is NP- complete problem. Exponential time complexity algorithm because the time-consuming, not applicable in real-time systems. Therefore, the lower time complexity and better algorithm are needed.

A service path delay ς_p is the sum of transmission delay and data on the link processing delay at a node. It is assumed that m is the original amount of data, so m $w.p_0$ represents the processing delay of data in SO.

$$t\left(\theta\right)=\begin{bmatrix}\cos\left(\theta\right)-\sin\left(\theta\right)\\ \sin\left(\theta\right)\cos\left(\theta\right)\end{bmatrix} \quad \textbf{(1)}$$

$$\ddot{z}=U\left(x\right)+m\left(x-U\left(x\right)\right) \quad \textbf{(2)}$$

In this format, $u_i.t_j$ represents the amount of data transferred to a certain service copy s_i.

$\left(\frac{u_i.t_j}{h_{i,j}}+u_i.t_j.p_j\right)$ represents the sum of transmission delay in a $link_{i,j}$ and the processing delay in s_j. The reliability t_j of a service path is the product of all service copies' reliability in this service path.

$$\begin{bmatrix}\varepsilon_i^{i+1}\\ \varpi_i^{t+1}\end{bmatrix}=t\left(\theta_i\right)\begin{bmatrix}\varepsilon_i^j\\ \varpi_j^i\end{bmatrix}=\begin{bmatrix}\cos\left(\theta\right)\\ \sin\left(\theta\right)\end{bmatrix}\begin{bmatrix}\varepsilon_i^j\\ \varpi_j^i\end{bmatrix} \quad \textbf{(3)}$$

The target is to make ξ_p smaller and t_j bigger. Optimization of the two targets is more complex, so it is proposed that the optimization objectives "ratio of delay and reliability" $\left(\frac{Delay}{\mathrm{Re}\,liability}, DIR\right)$ is taken as the goal of optimization.

$$\underset{|\vartheta\rangle\to|0\rangle}{\Delta\vartheta}=\varpi_{|0\rangle}-\varpi_{|\varphi\rangle}=-\arctan\left(\left|\beta\right|/\left|\alpha\right|\right) \quad \textbf{(4)}$$

The problem is defined as follows: In a given service graph SG (V, E) conditions, how to choose a path from S0 to St Service and make the D/R as large as possible. It will be expressed in the following content in detail that an approximate algorithm is proposed based on this definition, which is called as RC algorithm (lowest delay/reliability path), to solving the problem in polynomial time.

Functional Image and Service Image

Because for the same type of user's service, generally speaking, there are a variety of functions service combination can meet the demand. Therefore, the FG is used to represent the system the combination of all the possible functions of the service relationship. There are five basic assumptions system multimedia processing services, as shown in figure 1, then set off from f_0 to f_4 there are four possible ways of service combination, respectively

($f_0\ f_1\ f_4$, $f_0\ f_1\ f_3\ f_4$, $f_0\ f_1\ f_2\ f_4$, $f_0\ f_1\ f_2\ f_3\ f_4$) .

Function diagram describes the system function abstraction. Furthermore, each service has a plurality of service we consider all copies, copies of each service in the function diagram is extended form as shown in Figure 2 service graph (service graph, referred to as SG). Service description is the dependence between the copies of all services, contains all the information of the system. In Figure 2, assume that the service f_0 has 1 copy, 2 copies of f_1, f_2 has 1 copy, 2 copies of f_3, f_4 has 2 copies, and add a St as the final transfer of multimedia data the destination node to arrive. As we can see, f_4 f_0 f_1 f_2 corresponds to a path in FG, if you choose as a service instance of f_1, as a service instance of f_4, then in SG service path specific. At the same time definition, if there exists a directed edge from the $SOs_4^2, s_1^2 s2 s_4^2$, then s_j is called the precursor node s_j, s_j for subsequent s_i.

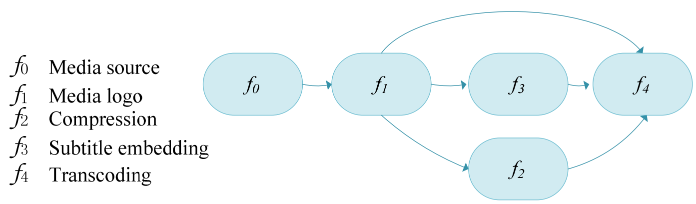


Figure 1 : Domain

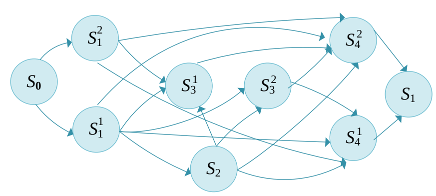


Figure 2 : Service area

RC ALGORITHM

The amount of data approximation

Some multimedia services will change the amount of data, so before the path was not determined, we cannot know a service node or link on the actual amount of data transmission. In Figure 2, a S0 data is m, so before the path is not determined, processing data may be from s_0 and $s_1^1(or\, s_1^2)$ transmission reaches, there may be from s_0 $s_1^1(or\, s_1^2)$ reach and s_2 transmission. Therefore, the data may be $m.r_0.r_1$ m·r0·r1. This uncertainty makes the problem become more complex, so it is needed to approximate amount of data processing.

Node Splitting

To solve the shortest path algorithm is the single source weights at the edge of the shortest path problem, but our service on the node graph have cost (weight), so we need to make some transformation of service graph, which can be used the shortest path algorithm.

Choi et al proposed method of node splitting to translate some network problems. This method is used. For example, in figure 2, we will s_2 split into two nodes $s_{2,1}$ and $s_{2,2}$, then let all the precursor node s_2 are connected to the $s_{2,1}$, $s_{2,2}$ connected to all nodes of S2. Data processing delay raw to s_2 node above, now use between $s_{2,1}$ and $s_{2,2}$ the cost of edge to said, it is called such as between $s_{2,1}$ and $s_{2,2}$ for the internal side edge (inner- link). Our service in every node splitting process, adding a new node, eliminating the vertices above cost, the price converted services diagram exists only on the edge.

Time Complexity

After the cost of labeling on service chart, you can use the shortest path algorithm to solve the problem. For the shortest path, node sequence we will produce a reduction to convert before service graph, paths can be obtained from RC. We'll let N_n and N_e respectively to represent a number of the number of nodes and edges the original service in the graph. The approximate second algorithms in data quantity, because each node at most connected with the N_n nodes, so the process of traversal of all edge up N_n times, the complexity of this part is $O(N_n N_e)$; in the third part the same each node is visited once, at the same time to find the all of its possible precursor node, this part of the complexity is $O(N_n N_e)$. The process of node splitting and data approximation are very similar, need to deal with each node, and then determine the precursor node and successor node need access to the N_e side, because of the complexity, this part of the $O(N_n N_e)$. In the reliability of conversion, since only Nn internal boundary, so the complexity is $O(N_n)$. The cost of annotation to deal with each edge, complexity is $O(N_e)$. The shortest path algorithm, we use the Dijkstra algorithm, its complexity is $O(N^2)$, wherein, n represents the number the vertices in the graph. According to the node the division process with $n \prec 2N_n$, at the same time, according to the rules of the complete graph $N_e \le N_n^2$ of RC algorithm, so the total time complexity.

$$\bar{t}(x,y) = t\left(x,U\left(y\right)\right) = \sqrt{\sum_{j=1}^{n} t\left(t_j\left(x\right)\right), U_j\left(t_j\left(y\right)\right)^2} \quad (5)$$

Reliability conversion

Due to the dynamic and mobility in pervasive environment, service replica node may fail. System requirements is to find a delay as small as possible, path of service and reliability as high as possible. The idea is to make the reliability of the parameters, the processing delay of the node.

The previous QoS related research work, a coefficient for each service internal edges in graph, the coefficient of reliability service node is an internal edge represents the inverse of E. After this treatment, the service replica node distribution coefficient of higher reliability coefficient is relatively small, low reliability of service replica node distribution the relatively large. These internal edge new delay cost is the original price multiplied by the respective coefficient, as processing delay internal edge new.

This process idea is straightforward: for the single source shortest path algorithm for the shortest path, a service copy reliability makes the low reliability of the inner side of the price is relatively high, the higher reliability of service copy internal edge cost becomes lower, and the shortest path algorithm to select the the shortest path selection bias in service copy the higher reliability. That is to say, the service replica node reliability higher more easily by the algorithm. The RC algorithm of time delay is small selected path, at the same time, high reliability, to achieve a balance between the two.

EXPERIMENT AND SIMULATION

Simulation Design

The simulation experiments on Inter (R) Core (TM) 2Duo CPU E8200 @ 2.66GHz, execute the 2GB RAM machine.

The simulation experiment using Java language, mainly includes the generation of various parameters services diagram and 3 kinds of algorithm. Parameters in the figure are designed to be adjustable, if we set the MaxNode=2000 service node. These service node, no multimedia processing service replicas initially, then according to the different network scale to produce functional diagram. For example, a MaxService=7 service in some settings, each service is at least MinInstance=3 copies, at most MaxInstance=7 copy, then we randomly generate a connected graph function, for each service between MinInstance and MaxInstance generates a random real copy number, and then in the MaxNode node randomly selected some nodes to deploy services. Service copy of compression ratio, the service node unit processing time and reliability, link bandwidth according to a normal distribution, then the service. We can through the MaxNode, MaxService, MinInstance and MaxInstance parameters to scale adjustment services graph, comparative experiments were carried out in different network scale. In the design of 3 algorithms, Random algorithm is easy to implement, Optima L algorithm dynamically apply the memory to hoRC all the possible path, RC algorithm according to the 4 steps in the previous section of the processing.

Evaluation Methods

There are two main experimental purposes: one is to evaluate the efficiency of RC algorithm, two is to evaluate the effect of selection algorithm. We have implemented two other algorithms to compare efficiency: one is random (random) algorithm, Random algorithm to select the service path, a successor node of the current node randomly selected as the processing nodes in the next step; two is the optimization (optimal) algorithm, Optimal algorithm traversing all possible paths, and then select the best path.

Service chart after the above 3 steps after the conversion, each side is marked time delay cost now. If a boundary is an internal edge, then it is the price of the following form.

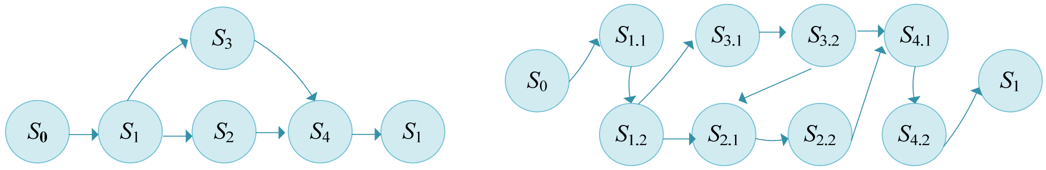
$$D_i = \sum_{i=1}^{k}\sum_{j=1}^{m}\left[t_i\left(k_x\right)\right]^2 \quad (6)$$

Among them, e_i and o_j are reliability and unit processing time of the internal edges represent service replica node, MI is the approximate amount of data; otherwise, this edge is the price of the following form:

$$\sum_{i=1}^{k} t_i\left(k_x\right) = 1, i = 1, 2, \ldots m \quad (7)$$

Among them, b_j is the service bandwidth, m_i is the service on the edge of the approximate amount of data transmission.

Figure 3 shows a simple conversion example. Figure 3 (a) is a service graph, assume that the only 1 copies of each service service (similar to handle multiple copies), node splitting, converted to figure 3 (b). s_0 and s_t shown is the node of source data and the destination node.



(a) Aservice graph (b) Transformed service graph

Figure 3 : Shows a simple conversion example

Considering the amount of data approximation and reliability conversion, assuming the initial data in the s_0 for the amount of M, then figure 3 (b) between $s_{1,1}$ and $s_{1,2}$ in the inner side of the price for $\frac{1}{e_1}m.r_0.o_1$, said the reliability of node s_1 data processing delay is multiplied by s_1; $s_{3,2}$ and $s_{2,1}$. Edge costs

$$D_i = \frac{\sum_{j=1}^{m}\left[t_i\left(k_x\right)\right]^2 t_i}{\sum_{i=1}^{m}\left[t_i\left(k_x\right)\right]^2}, i = 1,2,\ldots,n \quad \textbf{(8)}$$

Composed of transmission delay on the link between $s_{2,2}$ and $s_{4,1}$; on the side of price is as follows.

$$t_i\left(k_x\right) = \frac{\left(1-\left\|t_i-d_i\right\|^2\right)^{\frac{1}{(c-1)}}}{\sum_{k-1}^{c}\left(1/\left\|t_i-d_i\right\|^2\right)^{\frac{1}{(c-1)}}}, i = 1,2,\ldots,m, i-1,2,\ldots n \quad \textbf{(9)}$$

Because of this edge on the amount of data has two possible values (one is from s_0 through s_1 and s_3 transfer to s_2, two to s_2 from $s\, s_0$ by s_1 transmission), the cost of data approximation; $s_{4,1}$ and $s_{4,2}$ between the inner side of the price.

$$sk_i = \sum \frac{1}{t_i\left(s\right)}\left(s-t_i\right) \quad \textbf{(10)}$$

The internal side represents the service node has two precursor node, but also one of the precursor node still has two precursor node. So, in two the amount of data approximation.

Results of Analysis

The running time of 3 algorithms were compared in the topology of the network scale under different operation respectively. 3 kinds of algorithm 100 times, and then take the average execution time of 100 runs for comparison, as shown in Figure 4. We can see, the running time of RC algorithm and Random algorithm are much less than the optimal algorithm. When the network size is large, the running time of RC algorithm and Random algorithm are almost the same, therefore, the higher the efficiency of RC algorithm. In addition to the data shown in Figure , we also compared the running time of each algorithm is more large-scale network topology, node number reached 70, Optimal algorithm execution time was more than 80s. in a real-time multimedia transmission system, 80s delay is clearly not acceptable. So in the following experiments, when the number of nodes exceeds 70, we no longer consider optimal algorithm.

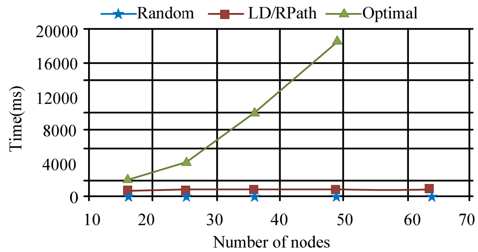


Figure 4 : The effect of 3 kinds of algorithms are compared in different network size

Then, the effect of 3 kinds of algorithms are compared in different network size. The experimental data shown in Figure 5 is the MaxService=7, MinInstance=2, MaxInstance=7 configuration. Under this configuration, we run 50 experiments in each experiment, the topological structure of network is to randomly generated, each experiment included 1 times 1 times RC algorithm, Optimal algorithm and Random algorithm 20 times, the results of the Random algorithm, the best results in 20, time delay, reliability and delay / reliable ratio as shown in Figure 5.

Figure 5 (a) that is the 50 set of experiments, comparing 3 delay path generation algorithm. We can see, the RC algorithm is very close to Optimal algorithm, are better than Random-Best. statistical information discovery. In a 50 experiment, a 44 RC delay path generated by the algorithm is smaller than the Random-Best algorithm, a the path with the Optimal algorithm to generate the same 24. Figure 5 (b) is the reliability comparison, in the 50 set of experiments, the reliability of route RC generated by the algorithm is 26 times more than the Random-Best algorithm, 1 have 6 times more than the Optimal algorithm. Figure 5 (c) is in contrast to delay / reliable ratio, delay / observe the path to produce RC and Optimal algorithm is very close to the reliable, are better than in most cases Random-Best algorithm. The results show that, in the 50 set of experiments, a Random-Best algorithm is better than the RC algorithm 42 times, 9 times better than Optimal algorithm.

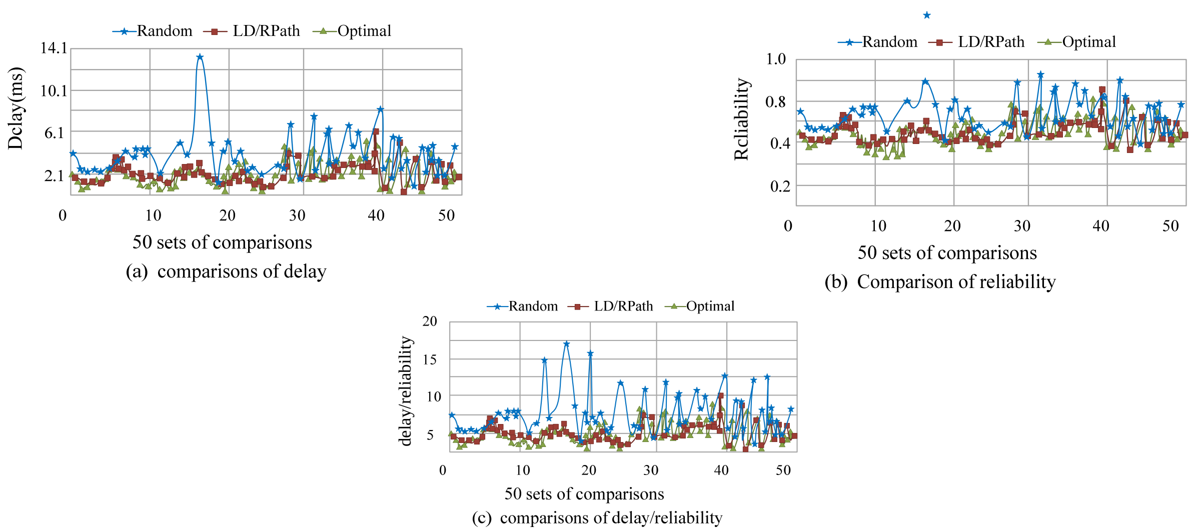
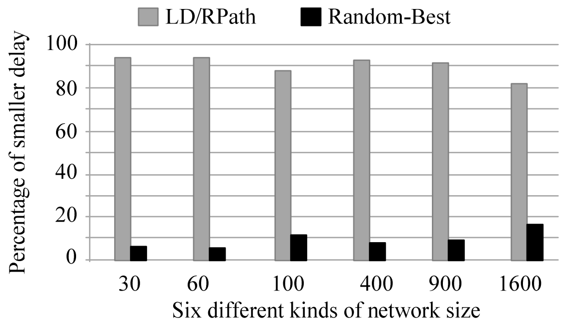


Figure 5 : 3 Kinds of algorithms the delay, reliability and delay / reliable than contrast

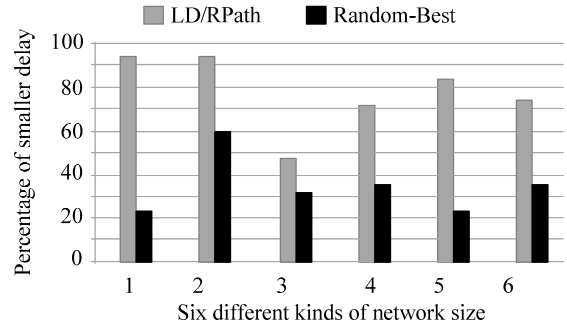
In addition, it is also compared the RC and Random algorithm in the node size selection effect of 30, 60100400900 and 1606 time (there is no comparison of Optimal algorithm, Optimal algorithm cannot because nodes in large scale to ensure real-time). In each node size, we run 100 experiments, each experiment consisted of 1 RC algorithm and 20 times of Random algorithm, Random algorithm results from Random-Best, the results as shown in Figure 6.

Figure 6 (a) expression in 6 different kinds of service network scale, 100 groups of experiments RC algorithm the number of generated path delay is smaller compared with the Random algorithm. We see, in the 6 case, the path delay were produced by RC algorithm 90 times, smaller than the Random-Best algorithm. We also observed to, as the network size increases, the performance of RC algorithm is slightly decreased, the size of the network for the 30 time, delay path RC algorithm produces is 94, smaller than the Random-Best algorithm; when the network size of 1600, times delay than Random-Best algorithm RC algorithm produces small drops to 82 times Fig. 6 (b) and 6 (c) respectively are compared both reliability and delay / reliable than. And Figure 6 (a) of the similar situation, we see the selection effect of RC algorithm is much better than Random algorithm.

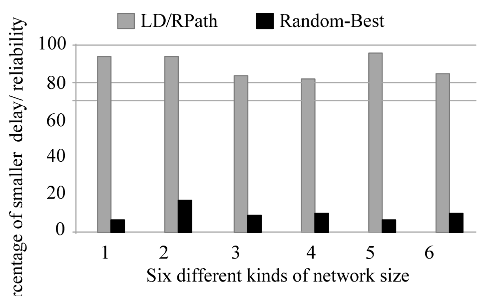
The two comprehensive efficiency and the selection effect of experimental results show that, RC algorithm in network topology is in small scale selection effect is close to Optimal algorithm, and running time almost equal to Random algorithm. When the network topology size become bigger, RC algorithm selection effect is better than the Random algorithm, Optimal algorithm cannot guarantee real-time gradually to be not applicable. In general, the RC algorithm with low time complexity to obtain good path selection effect, is suitable for real-time multimedia service oriented transmission.



(a)Comparison of higher reliability times



(b)Comparison of higher reliability times



(c)Comparison of Lower delay/ reliability times

Figure 6 : The RC algorithm and Random-Best algorithm in different network scale path selection effect comparison

CONCLUSION

Aiming at the problems in service oriented real-time multimedia transmission, such as a long delay, frequent jitter and low reliability, this paper proposes RC algorithm RC estimates the dynamic data volume on service nodes and links through reasonable data volume approximation. And the data splitting technique is imported to convert the node delay into the edge delay. In the mean time, the reliability of nodes is considered so that the multimedia delivery problem is transformed into a conventional shortest path problem. The innovations of this algorithm are as follows: the reliability is integrated into the edge weights reasonably, which guarantee the reliability in the choice of the shortest path; the reasonable approximate of data reduces the problem's complexity; node split integrates node weights into edge weights.

REFERENCES

[1] R.Berangi, S.Saleem, M.Faulkner, et al; TDD cognitive radio femtocell network (CRFN) operation in FDD downlink spectrum. IEEE, 22nd International Symposium on Personal, Indoor and Mobile Radio Communications, 482-486 **(2011)**.

[2] C.Kloeck, H.Jaekel, F.K.Jondral; Dynamic and local combined pricing, allocation and billing system with cognitive radios, The First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks (DySPAN), 73 – 81, **(2005)**.

[3] A.Kchiche, F.Kamoun, S.A.Makram, M.Günes; A traffic-aware infrastructure-based architecture for inter-vehicles file sharing, The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, UBICOMM, 44-49, **(2008)**.

[4] Hean Liu, Zhike Kuang, Cizhen Zhang, Goodson A. Yilly; Research on Spectrum Allocation Based on the Network Weights of Internet of Things, Journal of Networks, **8(12)**, 2804-2811 **(Dec 2013)**.

[5] K.C.Lee, S.H.Lee, R.Cheung, U.Lee, M.Gerla; First experience with CarTorrent in a real vehicular ad hoc network testbed, Mobile Networking for Vehicular Environments, 109-114 **(May 2007)**.

[6] A.Jangra1, Goel, N.Priyanka, K.Bhati; Security Aspects in Mobile Ad Hoc Networks (MANETs): A Big Picture, International Journal of Electronics Engineering, 189-196 **(2010)**.

[7] L.Buttyan, J.P.Hubaux; Stimulating Cooperation in Self Organizing Mobile Ad Hoc Networks, Technical Report No, DSC/2001/046, Swiss Federal Institute of Technology, Lausanne, **(August 2001)**.

[8] D.Niyato, E.Hossain, Z.HAN; Dynamics of multiple-seller and multiple-buyer spectrum trading in cognitive radio networks:A game-theoretic modeling approach, IEEE Transactions Mobile Computing, **8(8)**, **(2009)** http://dx.doi.org/ 10.1109/TMC.2008.157.

[9] An Chengquan, Liu Yang; A Matching Game Algorithm for Spectrum Allocation Based on POMDP Model, 7th International Conference on Wireless Communications, Networking and Mobile Computing (WiCOM),1-3 **(2011)**.

[10] Yuedong Xu, John C.S.Lui, Dah-Ming Chiu; On oligopoly spectrum allocation game in cognitive radio networks with capacity constraints, Computer Networks, **54(6)**, 925-943 **(29 April 2010)**.

[11] S.Antonatos, K.Anagnostakis, E.Markatos; Generating realis- tic workloads for network intrusion detection systems, ACM SIGSOFT Software Engineering Notes **29(1)**, 207–215 **(2004)**.

[12] M.Tavallaee, Cybernetics; Toward credible evaluation of anomaly-based intrusion-detection methods, Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on, 516-524 **(2010)**.

[13] J.N.Al-karaki, A.E.Kamal; Routing techniques in wireless sensor networks: a survey, IEEE Wireless Communication, **11**, 6–28 **(November 2004)** http://dx.doi.org/10.1109/MWC.2004.1368893.

[14] C.Li, M.Ye, G.Chen, J.Wu, An energy-efficient unequal clustering mechanism for wireless sensor networks, IEEE International conference on Mobile Adhoc and Sensor Systems Conference, 597-604 **(2005)**.

[15] Xin Huang, Xiao Ma, Bangdao Chen, Andrew Markham, Qinghua Wang, Andrew William Roscoe; Human Interactive Secure ID Management in Body Sensor Networks. Journal of Networks, **7(9)**, 1400-1406 **(2012)** http://dx.doi.org/10.4304/ jnw.7.9.1400-1406.

[16] Zhao Liangduan, Zhiyong Yuan, Xiangyun Liao, Weixin Si, Jianhui Zhao; 3D Tracking and Positioning of Surgical Instruments in Virtual Surgery Simulation, Journal of Multimedia, **6(6)**, 502-509 **(2011)**.

[17] C.Bfhm, S.BerchtoRC, D.A.Keim; Searching in High-Dimensional Spaces: Index Structures for Improving the Performance of Multimedia Databases, ACM Compute, **33(3)**, 322-373 **(2007)**.

[18] V.Paxson, R.Sommer, N.Weaver; An architecture for exploiting multi-core processors to parallelize network intrusion prevention. In: Sarnoff Symposium, 2007 IEEE. IEEE, 1–7 **(2008)**.