American J. of Engineering and Applied Sciences 2 (4): 620-627, 2009
ISSN 1941-7020

Error Resistant Real-Time Transport Control Protocol

S. Shahbazi, K. Jumari and M. Ismail
Department of Electrical, Electronics and Systems, Faculty of Engineering,
University Kebangsaan Malaysia, Bangi, Malaysia

Abstract: Problem statement: Real-time Transport Control Protocol (RTCP) protocol has been the subject of various criticisms due its problematic performance in large-scale networks. S-RTCP is a protocol with high potential as it has proved to be able to solve many problems of RTCP. It has numerous flaws on its own. This study aimed at dealing with flaws of S-RTCP and improving it in terms of stability and packet loss. **Approach:** A new proposed scheme was designed. Modifications included designing multi-manager scheme, improving parent-seeking procedures, reducing distribution of request packets, reforming the design to be independent from TTL, adding methods to check on sanity of manager nodes. This study considered packet loss ratio of below 2% as desirable. **Results:** ER-RTCP comparing to legacy RTCP in terms of packet loss using NS-2 in four different scenarios revealed improvements between 73 and 88% for various scenarios. It also kept packet loss rate below 2% for all scenarios. Comparison of ER-RTCP to S-RTCP showed that based on different α (stability of each single manager) values, ER-RTCP was more stable as it showed more resistance to entire scheme breakdown (β). ER-RTCP's parent-seeking procedure, as modeled scenario revealed a packet generation reduction of 97%, compared to S-RTCP's. In occurrence of parent AG leave or loss, ER-RTCP reduced request packet generation by 95%. Allowance of AG dismissing in ER-RTCP, avoided occurrence of packet loss, as sample scenario showed S-RTCP experiencing packet loss of 3.5% while ER-RTCP kept packet loss at zero in theory. **Conclusion:** Proposed design improved S-RTCP in terms of reduction of packet loss and stability.

Key words: RTCP, scalable-RTCP, hierarchy, stability, packet loss, TTL, network simulation, NS-2, error-sensitive

INTRODUCTION

Real-time Transport Protocol (RTP)[1] is a famous network protocol for real-time transportations. RTP is bundled with Real-time Transport Control Protocol (RTCP). RTCP is the control protocol of RTP, mainly responsible for tasks such as Quality-of-Service (QoS), adaptation, synchronization and so on. RTCP has proven to be a weak protocol when it comes to large-scale scenarios. Main problems associated with legacy RTCP are initial feedback flood, bye flood, excessive feedback delay, storage state and unnecessary RTCP RR packet reception.

Among the solutions proposed for RTCP's in-competencies, S-RTCP[2-4] has proved to be able to solve many of the problems associated with RTCP. However, S-RTCP introduces new problems on its own. The root of many of these problems is singularity of the manager. In S-RTCP, there are no limits set for the number of nodes that are allowed to become children of manager. This causes two major problems in large-scale sessions; congestion at links connected to manager and processing overload. First one is caused due to unlimited number of AGR packets sent to manager, while second one is due to heavy burden of processing a massive amount of AGR packets. Also, S-RTCP is heavily dependent on TTL field which, as shall be explained in detail, uses bandwidth sub-optimally. It also causes some incompatibility issues with some of the routing protocols. Finally, S-RTCP scheme is completely unstable to failures of manager as no precautions have been considered for circumstances that the manager fails for some reason such as an OS crash. Therefore, failure of manager simply results in failure of the entire S-RTCP scheme. This study makes an effort in improving S-RTCP by proposing series of reformations and new features, in terms of stability and packet loss.

Corresponding Author: S. Shahbazi, Department of Electrical, Electronics and Systems, Faculty of Engineering, University Kebangsaan Malaysia, Bangi, Malaysia

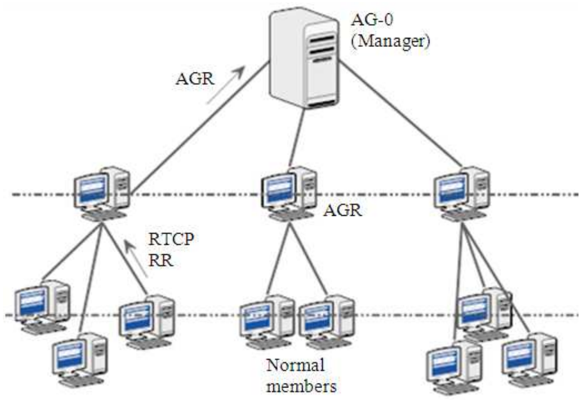

Fig. 1: Hierarchical structure of S-RTCP

Introduction to S-RTCP: S-RTCP applies a multi-level hierarchical architecture to RTP session in order to arrange RTP session participants and their reports[2-4]. S-RTCP uses hierarchy to restrain RTCP RR reports in smaller groups, as Fig. 1 shows.

In short, S-RTCP scheme claims to benefit from advantages compared to legacy RTCP such as: Elimination of storage state problem, feedback reports sent with minimal intervals, effective bandwidth usage, decrease in number of redundant packets, updated statistics for administrative purposes.

Disadvantages of S-RTCP: Hierarchical structure of S-RTCP includes normal members which are gathered in local regions, AGs and a single manager (AG-0) that AGs send summarized reports to it. AG-0 is the static center of the structure. Number of AG nodes that can connect to it basically is limitless, resulting in limitless AGR reports sent to AG-0 as well. Considering these, the weaknesses of S-RTCP are[6].

Congestion: In case numerous AGs become children of AG-0, congestion is likely to happen at AG-0 links, due to burst of AGR packets being sent to AG-0 regularly. This can result in loss of some AGR packets.

Processing overload: Considering above situation, another potential problem associated with burst of AGRs, is work overload on AG-0. Analyzing the statistics information sent by AGR packets involves mathematical processing which requires CPU processing power. Excessive number of AGR packets leads to CPU overload.

Error-intolerance: The design of S-RTCP is quite vulnerable to any kind of defects of AG-0. AG-0 is pre-assigned yet no precautions have been considered in S-RTCP design to let another node take over the responsibilities of AG-0. Therefore if some unexpected event occurs to AG-0 (e.g., a crash of operating system) and it fails, the S-RTCP session will fail as well since the scheme is entirely dependent on a single AG-0.

Dependency on TTL field: In order to be able to traverse inside the hierarchical tree of nodes e.g., perform ring searching[5], S-RTCP makes use of IP header's TTL field. TTL scoping[7], can be troublesome at certain situations. For example, it does not allow overlapping regions. Also it conflicts with some routing protocols such as Distance Vector Multicast Routing Protocol (DVMRP).

MATERIALS AND METHODS

ER-RTCP, contributes to error-sensitive real-time systems. ER-RTCP improves stability of the session while decreasing packet loss. These characteristics are especially welcome in real-time systems that have the least tolerance on packet loss, e.g., multi-player online gaming systems. In order to improve utilization of RTCP packets, a new scheme has been designed which has been inspired from S-RTCP, but comes with major improvements in many terms, especially stability and reduction of packet loss.

In ER-RTCP, as Figure 2 shows, nodes are virtually arranged in a hierarchical tree order. The tree arrangement is based on the type of each node within RTP session. In ER-RTCP, the height of the hierarchical tree has been fixed at three levels. This is unlike S-RTCP scheme in which the depth of the tree can be variable, resulting in generation of longer hierarchical tree, thus higher traverse times for packets.

At the top of hierarchical tree, which is level 0, is the main manager (main AG-0). At level 1, there are other AG-0s, including secondary AG-0. Level 2 of the tree consists of AGs and finally, the leaves of this tree are normal members. The session starts with main AG-0. Then other nodes join in. Secondary AG-0 can be pre-assigned or be chosen dynamically. However, under any circumstances, at least two AG-0s are to be present in the RTP session. ER-RTCP gives the joining members the option not to operate as AG in ER-RTCP. This option can come in handy for members connected using very low bandwidth links.

Entities in ER-RTCP: ER-RTCP consists of the following entities, some of which are newly included while some others already exist in legacy RTCP or S-RTCP.

Normal member: Any normal RTP session participant.

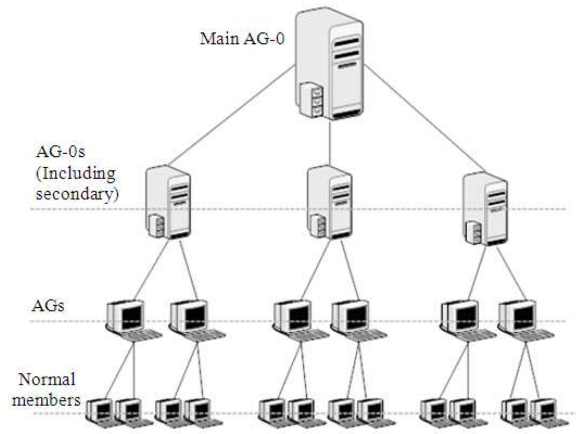

Fig. 2: The architecture of ER-RTCP

0 bits			8	16	24
V=2	P	RC	PT=208	Length	
SSRC of packet sender					
Children No			Number of RRs summarized		Max loss
Average packet loss (loss per node)					
SSRC of member with highest packet loss rate					

Fig. 3: AGR packet structure in ER-RTCP

Aggregator (AG): A session participant responsible for collecting RTCP RR packets sent by normal members (i.e. its children) within its scope. Based on the information within these packets, it creates summarized reports and sends them to its manager.

Aggregator Report (AGR): AGR is the new packet type which is used by AGs to send summarized information to managers. Note that AGR packet has been modified from S-RTCP version in order to serve the needs of the ER-RTCP. It contains information such as; Number of children, Total number of reports used for generation of the current AGR, Average packet loss, Maximum packet loss, SSRC of node with maximum packet loss. Figure 3 shows the structure of proposed AGR packet.

Manager (AG-0): It means AG of level zero. Since in the current design, managers are located at level 1 of the hierarchical tree (except for main AG-0), they should be called AG-1, but in order to keep basic things in harmony with S-RTCP, they are still referred to as AG-0.

The most obvious difference between ER-RTCP and S-RTCP's design is that S-RTCP uses one manager to handle the entire session, while ER-RTCP makes use of multiple managers. By assigning multiple managers to operate in the session, the new scheme becomes more immune from unexpected errors and failures. In ER-RTCP, two new entities have been created in the family of AG-0s; Main AG-0 and secondary AG-0. All AG-0s in the RTP session have to be checked for sanity on a periodic basis. This is the sole responsibility of main AG-0 to monitor sanity of other AG-0s and upon finding a faulty AG-0; it must take action by disposing it from its AG-0 status and replacing it with a new AG-0. On the other hand, the secondary AG-0 is solely responsible for monitoring sanity of the main AG-0. The procedure of looking up a new AG-0 is summoned whenever one of the following situations occurs: (1) The ratio of normal nodes to AG-0s exceeds a certain ratio. (2) An AG-0 reaches a very high CPU load threshold. This value again can be set by admin. (3) An AG-0 experiences an unexpected internal error e.g., crash of OS. Needless to say, this should be detected by other AG-0s.

Sender's and neighbor's list: All members have a list containing senders, AGs and AG-0s within the session, namely sender's list. Upon reception of either RTP data packets or sanity packets from a sender, the sender's list shall be populated. The list is populated with the ID of the sender, its AG status, as well as its traverse time. Normal members and AGs make use of their sender's list during the procedure of finding a parent. In addition to sender's list, all members have a list containing members at their vicinity, namely neighbor's list.

Independency from multicasting or broadcasting requests: In ER-RTCP, the RTP packets are utilized for sending AG status of the senders along with normal RTP data. Also in case the AG is not a sender, it broadcasts sanity messages to let other members be aware of its existence. When a session participant receives a sanity packet, it puts the ID of the sender, its AG status, as well as its traverse time, in its sender list. Later, this sender list shall be used for parent AG selection procedure.

Using this method, all the members will know about all the AG nodes active within the session with minimum packet distribution. By using ER-RTCP, several advantages are gained. First, there's no more need for parent-seeking nodes to broadcast 'search-for-parent' packets in order to find parent AGs. In addition to that, the problem of redundant reception of 'search-for-parent' multicasting is completely eliminated. Thirdly, this procedure is fulfilled much faster than S-RTCP's.

Table 1: An example of packet loss history

Report history	1	2	3	4	5	6	7	8
PL_i	3	0	0	1	0.0	2.0	1.0	0.0
W_i	1	1	1	1	0.8	0.6	0.4	0.2

Parent-seeking procedure: In S-RTCP, parent-seeking members send multicast requests in order to find a parent. No qualification procedure is included in the scheme as it simply picks up the first cooperative parent AG that it finds. It does not take into account the suitability of the candidate parent AG in terms of traverse time or packet loss rate. This leads to parent-seeking node choosing a sub-optimal parent AG.

In ER-RTCP, the parent-selection procedure first checks the current number of children of the potential candidate. If below the threshold, it then calculates the packet loss history of the potential candidate. Each member keeps track of packet loss rate of members within its neighbor's list and also sender's list during their last 8 transmissions. Then it performs the calculation by giving each field of packet loss history a different weight. The basis of this weighting is presented in[8]. Parent-selection procedure chooses the member with the lowest packet loss score. In case two or more potential candidates get equal scores (e.g., zero), the procedure uses their jitter value as the tiebreaker. Table 1 below shows an example.

The packet loss score is calculated using the following equation:

$$\sum_{i=1}^{8}(PL_i \times W_i)$$

Where:
PL_i = Stands for packet loss at field (i)
W_i = Stands for weight of field (i)

Using this equation, the packet loss score for this example would be:

$$(3\times1) + (1\times1) + (2\times0.6) + (1\times0.4) = 5.6$$

Adaptability: In S-RTCP scheme, after the parent-seeking member finds its parent AG, it sticks to the chosen parent AG throughout the entire session and does not look for a newer parent AG, except when forced to; i.e. when its parent AG leaves.

In case of a major change in network conditions, the scheme cannot adapt itself to the new situation. In ER-RTCP, each child is responsible to make sure its current parent AG is the best option available by performing periodic checks. The interval of this periodic check is set to 30 sec by default. This check utilizes the parent-seeking procedure. If the result of the procedure is equal to the current parent AG, nothing happens and the child keeps on cooperating with the current parent AG. But if the result of the procedure suggests a new parent AG, then the child sends a request to the new candidate parent and sends a bye packet to its current parent.

Dismission from serving the session: Unlike S-RTCP, ER-RTCP has taken into consideration that some members may not want to be considered as an AG candidate for reasons such as low-bandwidth connection (e.g., dial-up connection) that is barely enough for RTP packet reception. ER-RTCP gives the RTP session members the option not to operate as AG. This ensures that incapable members will be left alone to solely receive RTP packets.

Stability: The scheme provides immunity from scheme breakdown by starting with two AG-0s. Also in case of necessity, e.g., work overload, more AG-0s can be added to the managerial group. Also AG-0s monitor each other's sanity and in case of a failure detection, a new AG-0 is added to the group instantly. With the aforementioned precaution, the only way for ER-RTCP to fail is that all of the managerial nodes fail at once.

Comparison to legacy RTCP: For this comparison, a powerful and well-known network simulation tool entitled NS2[9] is utilized. Comparison of the two designs is done by inspecting the performance of the two designs in terms of packet loss, under various scenarios. Note that comparison of ER-RTCP to legacy RTCP in terms of stability is not applicable. This is due to the fact that legacy RTCP uses a distributed scheme and each member performs its tasks without interference of a superior entity. Simulation for each scenario is run 5 times. The final version of NS2 is chosen for simulation, which is version 2.33.

For first scenario, a tree-based topology is used, as shown in Fig. 4. There's only one sender, located at the root, while other 12 nodes are receivers. The send rate is set at 500 Kbps and the simulation lasts for 100 sec. Note that this duration is a desirable duration, used in many studies involving NS2.

For second through fourth scenarios, a transit-stub topology is used. As Fig. 5 shows, Transit-stub topology is hierarchical-based, fixed at two levels. It includes transit domains and stub domains, in which transit domains are at the root while stub domains are the bottom, owned by transit domains. The aforementioned topology is chaotic, providing a good resemblance of an unexpected, random topology.

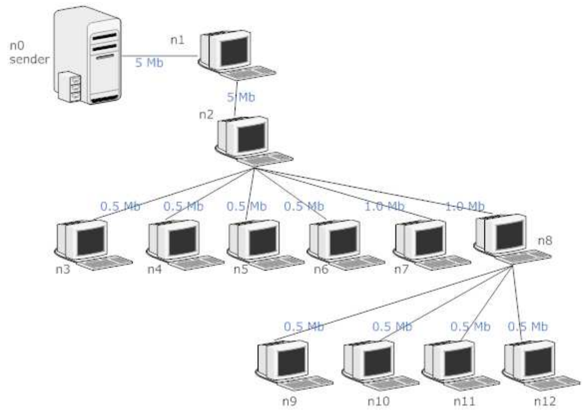

Fig. 4: The tree-shaped topology for 1st scenario

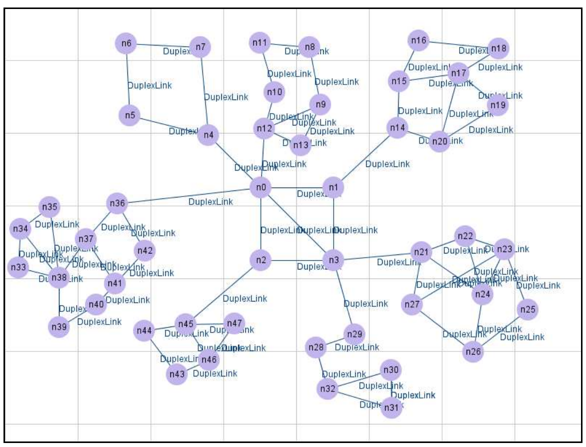

Fig. 5: Transit-stub topology used for second through fourth scenario

Transit-stub topology is a popular topology, used by many researchers in their network simulations. The topology consists of 48 nodes. It has one transit domain containing nodes 0 to 3, each of which has one or more stub domains attached to it. For second scenario, there is a single sender (node 0) and the remaining nodes are receivers. All the links within the topology have the similar bandwidth of 1 Mbps. The transmission rate is also set at 1 Mbps. The simulation is run for 100 sec. For third scenario, same transit-stub topology is utilized. But this time a total of 8 nodes participate in TCP traffic transmission. TCL nodes are chosen in a way that their traffic does not traverse in isolated parts of the network. Each TCP node's send rate is 0.2 Mbps. A single RTP sender is present with send rate of 1.0 Mbps. Finally in fourth scenario, multiple senders broadcast simultaneously, resulting in higher chance of packet loss. Thus, adaptability of the scheme in practice is important for good performance.

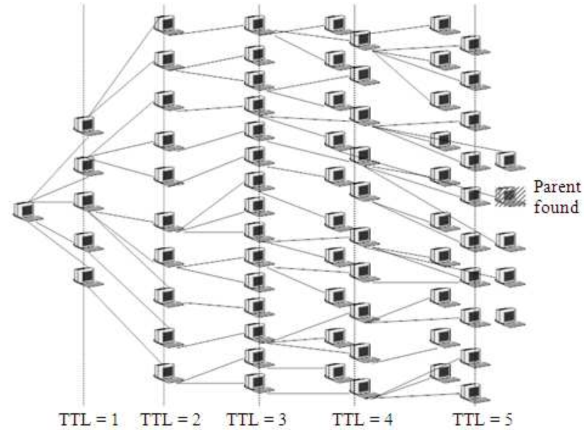

Fig. 6: Sample topology as viewed by the parent-seeking member

Each sender sends data packets with data rate of 0.2 Mbps. No non-RTP nodes exist. Each link has the capacity of 1 Mbps.

Comparison to S-RTCP: For this purpose, the two protocols shall be compared in terms of stability and traffic-generation of their algorithms. The results of these comparisons shall be presented in results.

Parent-seeking procedure: To clarify the difference in performance of parent-seeking procedures of the two schemes, the scenario shown in Fig. 6 is utilized. In ER-RTCP, the parent-seeking member finds its desired candidate parent before sending requests, therefore it unicasts its request to the desired candidate parent. For S-RTCP, equation below is used, which takes into account the broadcasting of 'search-for-parent' requests with increasing TTL values:

$$R = \sum_{i=1}^{T}\sum_{j=1}^{i}(N_j)$$

Where:

N_j = Number of nodes at hop distance of j from the parent-seeking node

T = TTL value at which the parent is found

R = Total request packets generated during search-for-parent procedure

Dismission from AG candidacy: In S-RTCP scheme, every normal member may be requested to serve as AG mandatorily. ER-RTCP gives members the option to dismiss themselves from this. Example below clarifies the outcome.

Consider an RTP session operating with transfer rate of 54 Kbps, utilizing S-RTCP. Two senders send audio and video streams. A member with a 56 Kbps connection (dial-up) has been assigned as AG. It has 20 children, each sends RTCP RR packets to this parent every 5 seconds. In addition to that, each session participant, including this parent, receives RTCP SR packets from senders every 5 sec. Packet size of RTCP SR and RR packets are 136 and 116 bytes respectively[1]. Reception of RTCP feedback traffic for this parent per second can be calculated using this formula:

$$\frac{(PS_{RR} \times N_{Children}) + (PS_{SR} \times N_{Senders})}{5\text{ sec}}$$

Where:
PS_{RR} = Packet size for RTCP RR
$N_{Children}$ = Number of children for the parent
PS_{SR} = Packet size for RTCP SR
$N_{Senders}$ = Number of senders in the session

Using this formula, feedback traffic for S-RTCP in this scenario is 4.04 Kbps. Adding this to RTP traffic reception (54 Kbps) results in 58 Kbps of incoming traffic which surpasses the parent's link capacity of 56 Kbps. However, in ER-RTCP, when being dismissed from AG status, they receive RTCP SR packets but no RTCP RR packets. Using the above formula, feedback traffic for ER-RTCP in this scenario is 435 bits per second; resulting in 54.4 Kbps of incoming traffic.

Stability: In S-RTCP scheme, the single AG-0 of the scheme does not have the option of leaving the session. However, it may crash or fail due to an unexpected event. In such cases, there are no measures considered by the scheme to replace the failed AG-0 with a new one. Therefore, scheme breaks down. ER-RTCP has taken this issue into account as the following example clarifies. Let's consider $\propto$ as the probability of a manager working flawlessly throughout an RTP session and β as the probability of entire scheme failure. Probability of entire scheme failure in S-RTCP is simply calculated using the following equation:

$$\beta = 1 - \propto$$

However, in ER-RTCP, there are at least two managers present. Therefore, for a scheme failure to happen, in worst case two managers must fail at the same time. Considering n as the number of managers, the probability of such event may be calculated using equation below:

$$\beta = (1 - \propto)^n$$

For example, considering $\propto = 0.98$, β for S-RTCP is 0.02, while for ER-RTCP (considering n = 2), β would be 0.0004.

Situation under which a parent AG is gone: Leaving or sometimes failing of a parent AG in RTP session happens occasionally. Parent-seeking procedure plays a major role here as it's called numerous times in this regard. In S-RTCP, when a parent is gone, each of its children has to broadcast 'search-for-parent' messages. Upon not finding a parent, TTL value is increased and 'search-for-parent' is repeated until a parent is found.

For example, consider the RTP session in which session contains 50 members, out of which an AG with 12 children has suddenly crashed or left the session. Each node has an average of 4 nodes at its hop distance of 1, 10 nodes at hop distance of 2 and 14 nodes at hop distance of 3. In the end, 2 of the nodes find their parent at hop count of 1, 7 nodes find their parent at hop count of 2 and the remaining 3 find their parent at hop count of 3.

In S-RTCP scheme, the amount of requests sent during this incident is calculated using equation below:

$$R = \sum_{c=1}^{n} \sum_{i=1}^{T} \sum_{j=1}^{i} (C_{N_j})$$

Where:
C_{N_j} = Number of nodes at hop distance of j from the parent-seeking node for child C
T = TTL value at which the parent is found
n = Number of children that lost their parent
R = Total request packets generated by children of the lost parent

Using equation, the total request packets generated for the aforementioned examples is calculated as follows:

R = [2 children × 4 nodes] +
[7 children × ((4)+(10+4)) nodes] +
[3 children × ((4)+(10+4)+(14+10+4)) nodes] = 272

In ER-RTCP, the amount of request packets sent in the similar situation is: (12 × 1) = 12. This is due to the fact that ER-RTCP does not need to multicast request and instead it sends a unicast request to its desired parent.

RESULTS

In regards to comparisons between ER-RTCP and legacy RTCP, the statistics obtained from simulations using the four aforementioned scenarios are presented. Table 2 provides the results of simulations of 1st scenario in which ER-RTCP shows packet loss reduction of 73%. In 2nd scenario, as Table 3 reveals, a packet loss reduction of 70% is achieved. In 3rd scenario, shown in Table 4, ER-RTCP reduces packet loss by ratio of 68%. Finally Table 5 shows in 4th scenario, ER-RTCP reduces packet loss by 88%.

In regards to comparisons between ER-RTCP and S-RTCP, Fig. 7 shows the result obtained from comparing the parent-seeking procedure of ER-RTCP and S-RTCP.

The model presented in regards to dismission of AG candidacy feature shows that in ER-RTCP, AG node receives 54.4 Kbps of feedback traffic while S-RTCP receives 58 Kbps. Considering model's 56 Kbps link, S-RTCP suffers from packet loss rate of at least 3.5%, while E-RTCP does not surpass the link capacity thus keeping packet loss at zero, in theory.

Probability of scheme breakdown for the two schemes according to different values of $\propto$ is shown in Fig. 8. It shows that in response to growth of chances of failure of each manager, ER-RTCP provides better stability compared to S-RTCP.

In occurrence of parent AG leaving session, scenario shows that ER-RTCP reduces generation of request packets by ratio of 95%, compared to S-RTCP.

Table 2: Results of simulation using 1st scenario

	Legacy RTCP	ER-RTCP
Packets expected	73747.00	73998.00
Packets lost	4880.00	136.00
Loss rate (%)	0.66	0.18

Table 3: Results of simulation using 2nd scenario

	Legacy RTCP	ER-RTCP
Packets expected	576524.00	579248.00
Packets lost	14071.00	4288.00
Loss rate (%)	2.44	0.74

Table 4: Results of simulation using 3rd scenario

	Legacy RTCP	ER-RTCP
Packets expected	477282.00	479749.00
Packets lost	12652.00	3967.00
Loss rate (%)	2.65	0.83

Table 5: Results of simulation using 4th scenario

	Legacy RTCP	ER-RTCP
Packets expected	574850.00	576842.0
Packets lost	30309.00	3446.0
Loss rate (%)	5.27	0.6

DISCUSSION

As Fig. 9 suggests, ER-RTCP shows better performance in terms of packet loss, compared to legacy RTCP. It keeps packet loss rate below 2% goal threshold in all scenarios. Also, ER-RTCP shows a more diverse performance compared to legacy RTCP, as network complexity increases. Additionally, results suggest that ER-RTCP performs better in scenarios that traffic is dedicated to RTP traffic. Comparing to S-RTCP, ER-RTCP improves stability of session by showing more resistance to scheme breakdown. It's remarkable that stability of ER-RTCP session further increases as more managers are added to the managerial circle. Dismission of AG candidacy feature contributes to less packet loss rates in low-bandwidth links.

Studying the behavior of the two protocols during parent-seeking procedure and occurrence of parent AG leaving, suggests that ER-RTCP highly reduces traffic generation of these procedures, directly resulting in less traffic and therefore less packet loss. It also noticeable that reduction rate becomes higher as session becomes

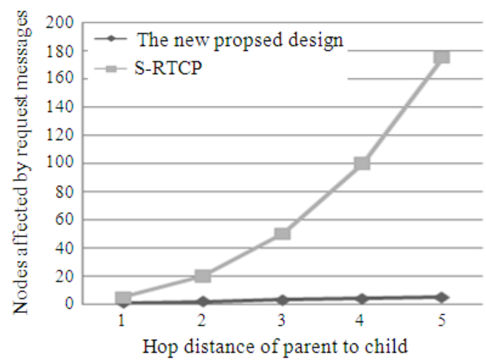

Fig. 7: Number of nodes affected by request messages in S-RTCP and ER-RTCP

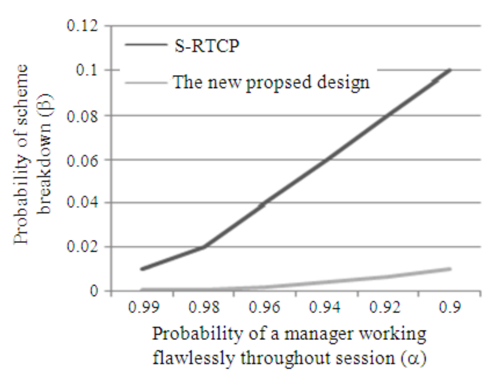

Fig. 8: Probability of scheme breakdown for S-RTCP and ER-RTCP, according to different $\propto$ values

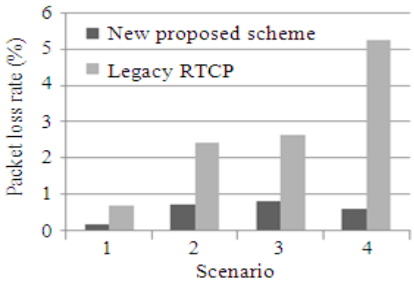

Fig. 9: Comparison of packet loss rate between legacy RTCP and ER-RTCP in different scenarios

larger in scale. To sum up, it can be claimed that ER-RTCP accomplishes its goals of improving stability and reducing packet loss.

CONCLUSION

This study aimed at reforming S-RTCP by improving it in terms of stability and packet loss reduction. For this purpose, various modifications and new features were proposed. ER-RTCP outperformed legacy RTCP on all of the simulations. Results revealed improvement rates between 73 and 88% in regards to packet loss, for different scenarios simulated. Comparing to S-RTCP in terms of stability, ER-RTCP proves more stable. Also, ER-RTCP reduced its traffic generation in parent-seeking procedure by rate of 97%, as modeled scenario showed. In situation of parent AG loss/leave, sample scenario showed decrement rate in traffic of request packets of 95%. ER-RTCP allows low-bandwidth members to be dismissed from serving as AG, preventing occurrence of packet loss in such members. Calculations based on example scenario showed S-RTCP scheme, by assigning low-bandwidth members as AGs, results in packet loss of 3.5%; while in theory, ER-RTCP kept packet loss at zero.

REFERENCES

1 Schulzrinne, H., S. Casner, R. Frederick and V. Jacobson, 2003. RTP: A transport protocol for real-time applications. IETF RFC 3550. http://www.ietf.org/internet-drafts/draft-ietf-avt-rtp-new-12.txt

2 El-Marakby, R., 2000. Design and performance of a scalable real time control protocol: Simulations and evaluations. Proceedings of the Fifth IEEE Symposium on Computers and Communications, July 3-6, IEEE XPlore Press, Antibes-Juan les Pins, France, pp: 119-124. DOI: 10.1109/ISCC.2000.860623

3 El-Marakby, R. and D. Hutchison, 1998. A Scalability Scheme for the Realtime Control Protocol, Proceedings of the IFIP TC-6 Eigth International Conference on High Performance Networking¸ Sept. 21-25, ACM Press, The Netherlands, pp. 153-168. http://portal.acm.org/citation.cfm?id=743049

4 El-Marakby, R. and D. Hutchison, 2003. Scalability improvement of the real-time control protocol. Comput. Commun. 28: 136–149.

5 Yavatkar, R., J. Griffioen and M. Sudan, 1995. A Reliable Dissemination Protocol for Interactive Collaborative Applications. Proceedings of the third ACM international conference on Multimedia, ACM Press Nov. 5-9, San Francisco, California, United States pp. 333-344. http://portal.acm.org/citation.cfm?id=215288

6. Shahbazi, S., K. Jumari and M. Ismail, 2009. A new design for improvement of scalable-RTCP. Proceedings of the International Conference on Future Computer and Communications (ICFCC'09) Apr. 4-5, Kuala Lumpur, Malaysia, pp: 594-598.

7 Meyer, D. 1998. Administratively Scoped IP Multicast. IETF RFC 2365. http://www.isi.edu/in-notes/rfc2365.txt

8 Bouras, C., A. Gkamas and G. Kioumourtzis, 2007. A framework for cross layer adaptation for multimedia transmission over wired and wireless networks. Proceedings of the 2007 International Conference on Internet Computing Jun. 25-28, Monte Carlo Resort, Las Vegas, Nevada, USA. http://ru6.cti.gr/ru6/publications/2739ICOMP2007_Bouras.pdf

9 Mahrenholz, D. Ivanov, S. 2004. Real-time network emulation with NS-2. Eighth IEEE international symposium on distributed simulation and Real-time Applications. Proceedings of the 8th IEEE International Symposium on Distributed Simulation and Real-Time Applications, Oct. 21-23, IEEE Xplore Press, USA., pp: 29-36. DOI: 10.1109/DS-RT.2004.34