IA-FEC: INTERLEAVED ADAPTIVE FORWARD ERROR CORRECTION SCHEME WITH EFFICIENT PACKET LOSS RECOVERY SYSTEM IN CLOUD COMPUTING

Benjamin Franklin I[1] and Ravi T. N[2]
[1]Department of Computer Applications, St. Joseph's College of Arts and Science, Cuddalore, Tamil Nadu, India
[2]Department of Computer Science, Periyar E. V. R. College, Trichy, Tamil Nadu, India
E-Mail: franklinbenj@gmail.com

ABSTRACT

Cloud computing plays an important role in next generation of business enterprises. In traditional method of IT services maintained their data under suitable physical, logical and personnel controls, but in cloud computing large data centers consisting application software and databases, where the organizational data and services may not be fully reliable. This paper focuses on data storage security against packet loss which is most important quality of services for the cloud users. Some existing techniques have been introduced in recent years against packet loss either by associate with retransmission request by cloud users or adding redundant Forward Error Correction (FEC) data. This paper proposes an efficient data encoding method of Interleaving Adaptive-FEC (IA-FEC) method for time sensitivity of data recovery named as Interleaving-A-FEC has been introduced. By use of interleaving technique, the opportunity to recover lost packets can be much improved due to the interleaving characteristics to separate the effect of packet losses. Adaptive–FEC (A-FEC) has the advantages of high redundancy rate with respect to the packet loss, based on the request messages received from the cloud users. This method combined both the advantage of A-FEC and interleaving techniques. The simulation results demonstrate that the proposed IA-FEC method has produced higher recovery rate than traditional FEC method, which is widely being used for internet phone services. The proposed approaches will be more suitable for bursty packet losses with various loss environments.

Keywords: adaptive forward error correction, bursty packet loss, cloud computing, data storage security, FEC, interleaving, redundancy rate, recovery rate.

1. INTRODUCTION

Cloud technology is a present trend in internet based development and computer technology. Software as a Service (SaaS) is an important computer architecture, which consist of both cheaper and powerful processor for transforming data centers into pools of computing service on a huge amount. The growing bandwidth requirement and trustworthy with flexible network connections create it even promising that users may now subscribe improved quality of services from data and software that reside exclusively on remote data centers. Data sharing in the cloud caused unlimited accessibility to the users as they do not have to care about the difficulties of direct hardware management. Familiar Cloud service providers are Amazon Simple Storage Service (S3) and Amazon Elastic Compute Cloud (EC2) [1]. These online services are providing large amount of storage space and organized computer resources for computing platform shift meanwhile neglecting the local machine responsibility of data maintenance in physical devices. Accordingly, users need to check with their service provider in terms of reliability and availability for their data [2].

In order to ensure the integrity and availability of data in cloud and implement the quality of cloud storage service, effective approaches that allow on-demand data precision verification on behalf of cloud users have to be implemented. Nevertheless, the point that users don't have any physical control of over the data in the cloud bans the direct implementation of convolutional cryptographic primitives for the determination of security in data integrity [3]. Therefore, the authentication of cloud storage precision must be directed without clear information of the data files [3-6]. Today, the major aspect of controlling the quality of service (QoS) of multimedia applications initiates from the variations in available bandwidth, delay and packet loss on wireless connections over cloud.

Retransmissions and FEC are used to avoid the packet loss in wireless data transmission. In the first methodology, copies of lost or corrupted packets are retransmitted upon getting negative acknowledgments, or afterward a timeout during which a positive acknowledgment flops to reach [7, 8]. Nevertheless, retransmissions will disturb the flow of the data stream, unless an adequately huge buffer is provisioned at the client. Furthermore, each retransmission demand must be controlled independently by the server, resulting in significant overhead.

FEC is an error-resilience technique that removes the retransmissions by sending the redundant information together with the original data to the client in a systematic way. This allows the receiver to reconstruct the original data without requiring any intervention from the server. This avoids the round trip delays of the retransmission requests. The main challenge in configuring the FEC systems is to determine the optimal ratio of the amount of redundant data sent along with the number of original data. Higher redundancy rates decrease the longer packet loss and/or frequent packet losses, but resulting in the higher bandwidth consumption. Lower redundancy rates preserve more bandwidth, but might not provide sufficient data to reconstruct all corrupted or lost packets. In an A-FEC, larger group sizes results in the longer reconstruction delays at the client and higher penalties in terms of redundancy rate when a feedback message is lost. Higher redundancy rate results in the increase of the bandwidth consumption [9].

In the wireless networks, there are more chances for the occurrence of burst packet losses. The burst packet losses in the communication networks cause the loss of information in the network and degrade the quality of voice by disturbing the speech patterns available from the received packets [10, 11]. Adaptive FEC mechanisms are developed to reduce the packet loss rate. However, A-FEC mechanisms cannot add the redundancy to recover the lost packets if the available bandwidth is inadequate. The burst packet loss reduces the recovery rate of the FEC mechanism, if the length of burst packet loss is greater than the length of FEC redundancy. To solve this issue, A-FEC is combined with the interleaving mechanism for data transmission through wireless networks.

In this article, an Interleaving A-FEC (IA-FEC) system that vigorously alters the redundancy rate to the packet loss perceived in the data stream is proposed based on the feedback suggestion received from the users. Packets are reasonably clustered, and the user informs the server immediately it has received adequate (original or redundant) data to rebuild the complete cluster. The server will stop the transmission of remaining packet of that cluster immediately. The projected technique allows reconstructing the data stream under different levels of packet loss, while reducing the additional bandwidth required for the redundant data and without negotiating on the QoS. The IA-FEC system is industrialized for multimedia applications that are distributed over a (typically wireless) connection that is subject to large variations in packet loss.

The article is organized as follows. Section II addresses the related works on IA-FEC to determine the merits and demerits of each research work. The development and implementation work of the proposed IA-FEC method are discussed in Section III. The results and discussions are presented in Section IV. Section V summarizes and conclusions regarding the IA-FEC functionality.

2. RELATED WORKS

Wang *et al.* [3] suggested an efficient and flexible distributed system with clear dynamic data maintenance to guarantee the accuracy of user data in the cloud. Their work is extended to user to review their cloud storage with easiest file communication and low computation cost, proposed method that is extremely effective and strong against malicious data alteration attack, and collusion attack in server. Wang et al. [12] introduced removal of modifying code in the file distribution training to deliver redundancies and ensure the data dependability. This structure might extremely decrease the communication and storage overhead as compared to the convolutional replication-based file distribution methods. Their system attains the storage accuracy as well as localization of data error, that is, if data corruption has been identified during the verification process, their method can almost assure the simultaneous identification of error detection.

Shah *et al.* [13] presented an effective model for data integrity in the cloud based on the Proof of Retrievability (POR) technique. Their model combines two techniques of spot-verification and error-correction code to ensure both control and retrievability of files on cloud service systems. Shacham and B. Waters [14] constructed random linear function based homomorphic authenticator for improved security with less communication link. An improved model for POR protocols are invented in the work developed by Bowers et *al.* [6], which is an improved version of [13, 14]. Sohn *et* al. [15] developed double cross-layer strategies to improve the video quality using the adaptive packet level FEC over IEEE 802.11 networks. Tsai *et al.* [16] proposed FEC with Interleaving mechanism combining Cognitive Technology for video streaming over the wireless networks.

Yang *et al.* [17] devised a new data error detection approach that discovers the computational capacity of the cloud environment and Wireless Sensor Network (WSN), based on the scale-free network topology. All existing techniques are concentrated on the standard data. The user security was checked before outsourcing the data files from the cloud. If any changes occur in the user content, then it is undergone through the error correction code system. However, this scheme suffers from high computational complexity and communication needs [6].

FEC has been well described in [18-20]. Also, these techniques are favored for the random data losses, wherever the perfect recovery is not assured. But, it has a limitation of overhead incurred due to the redundancy information. Though, there is an additional overhead of the FEC, its capability to correct the minor data losses in the error-prone environments such as wireless medium.

Interleaving is the desired resolution to improve the effects of burst errors and has been popularly implemented in the end-to-end multimedia applications [21-23]. A sender inserts the data packets before sending them out and the data packets has been reconstructed in received side with certain amount of delay. Level of interleaving is the term used to estimate the latency, meanwhile reconstructing of the interleaved data packets has required the exact delivery of all packets involved in the interleaving process.

3. PROPOSED WORK

Proposed framework for cloud packet loss by an efficient recovery model of interleaving based A-FEC method is illustrated in Figure-1. Figure-2 shows the flow diagram of the proposed work.

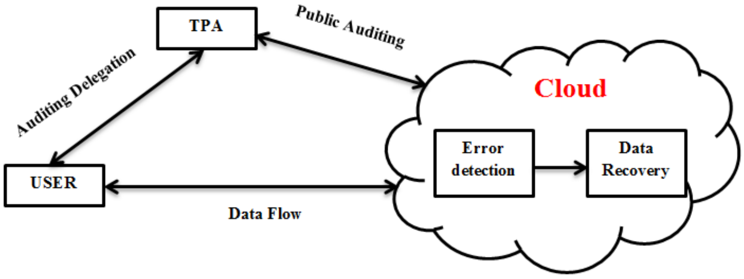


Figure-1. System architecture of the IA-FEC system.

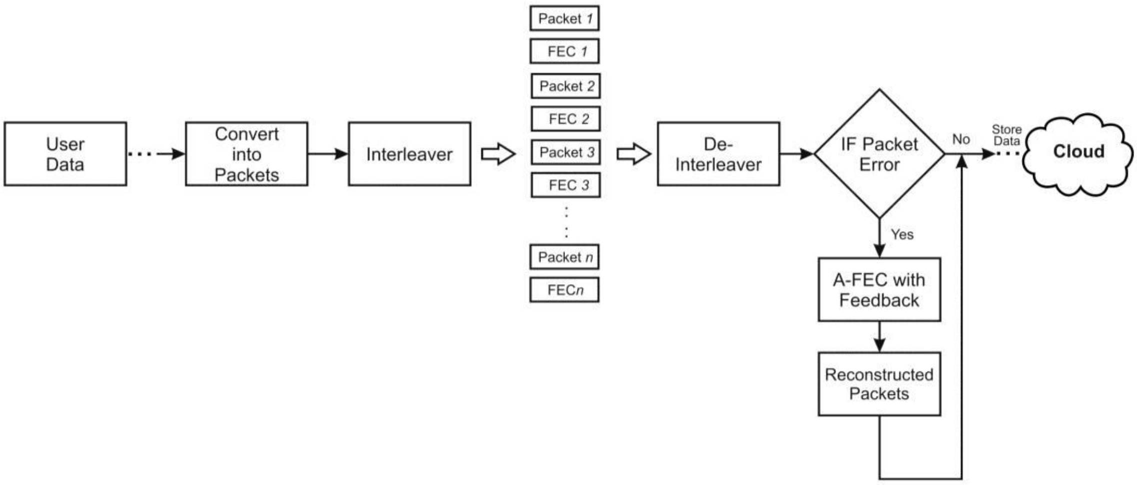


Figure-2. Flow diagram of the proposed IA-FEC system.

A. System model

IA-FEC is characterized by three entities such as

- User
- Cloud Service Provider (CSP)
- Third Party Auditor (TPA)

The proposed architecture has three vital entities,

User: User is an entity who can store their data in the cloud and can change their data stored in the cloud server. This can be an individual or organization.

CSP: This entity is responsible for managing the distributed cloud storage to online cloud access and it contains the storage resources too.

TPA: TPA is a trusted entity to access the cloud storage and prevent unauthorized access by received the request from the users. A one more special entity is designed to ensure the security of proposed method is Adversary Model. This model can constantly changing the data files stored on individual servers. Once the data is stored on server, adversary can modify the data by adding unwanted content or changing the original data [2].

They can store their secured data on cloud storage system and it is no longer can access locally. Hence, the data availability and integrity must be guaranteed by all distributed cloud servers. All data authorization can be effectively identified by using security system with proper error detection and correction techniques.

TPA is necessary to evaluate, audit and expose the risk of the cloud storage services. FEC encoders are normally parameterized with (m,k) tuple. For each outbound sequence of data packets, a total of (m+k) data and error correction packets are transmitted over the channel, while resulting in an encoding overhead of k/m. The redundant information cannot be generated and sent until all the data packets are available for transmission. Consequently, the latency of packet recovery is determined by the data transmission rate. Generation of the error correction packets lesser than the data packets at the sender is not a feasible option even though the data rate in this channel is low, the receiver and/or network could be operating at near full capacity with data from other senders. FEC is highly susceptible to the bursty packet losses.

B. Data error detection and correction

Error correction codes are both ineffective and inadequate in tremendously poor channel conditions. A recovery mechanism improves performance of transmission when error correction techniques are not efficient. Methods, that apply error corrections merged with retransmission as a recovery, are called Hybrid method of interleaving A-FEC. A-FEC is a packet level

FEC method. In this A-FEC technique, the number of redundant packets is continuously changed to send only the minimal number of redundant packets that is required to reconstruct missing packets. As well as usage of interleaving technique, the probability to recover lost packets can be much improved due to the interleaving features to decrease the effect of packet losses.

But, if the entire data packets were interleaved, the latency would become main issues as per the increase of interleaved packets. Therefore, Interleaving A-FEC merges together the strong point of both the FEC and interleaving techniques. It goals to combine the robustness of FEC against random errors, in addition to the interleaving's capability, to improve the effects of burst errors. The advantage of the constructed in FEC support is applied in the cloud technology.

C. Data recovery

a) Interleaving

FEC information is added to the storage devices for recovering the corrupted data. The data redundancy allows the receiver to detect the number of errors in the message and correct these errors without the need for requiring data retransmission. The FEC faces two challenges such as rate sensitivity and burst susceptibility. Interleaving is an encoding technique used for resisting the bursty packet loss, where the error correction packets are generated from the alternative separate blocks of data rather than from the consecutive packets. This technique adds burst tolerance to the FEC, but increases its sensitivity to the transmission rate.

b) A-FEC

As described above, using a Vandermonde matrix, the first 'K' packets are same to the original data. In the rest of this paper, the term original packets will be used for both the native data as for the first K generated data packets. The shared F redundant packets of a cluster, called FEC packets in this article, are stopped once a feedback message from the user is received. Subsequently, Reed-Solomon error correction codes are used, the user will send the feedback message for a particular cluster when it has received K packets of that cluster and only a subset $f \leq F$of the generated redundant packets are actually transmitted. The proposed A-FEC framework can be used to enhance both the number of transmitted FEC packets 'f' and Δ that represents the delay between the transmission of last original data packet and the first FEC packet of same cluster. Assumed the values of K and packet loss probability p, A-FEC can be used to choose the optimal value for f and Δ. In this article, f is constantly adjusted through feedback approach, while Δ is set by choosing the FEC group size and the interleaving pattern. Figure-3 illustrates the principle of A-FEC.

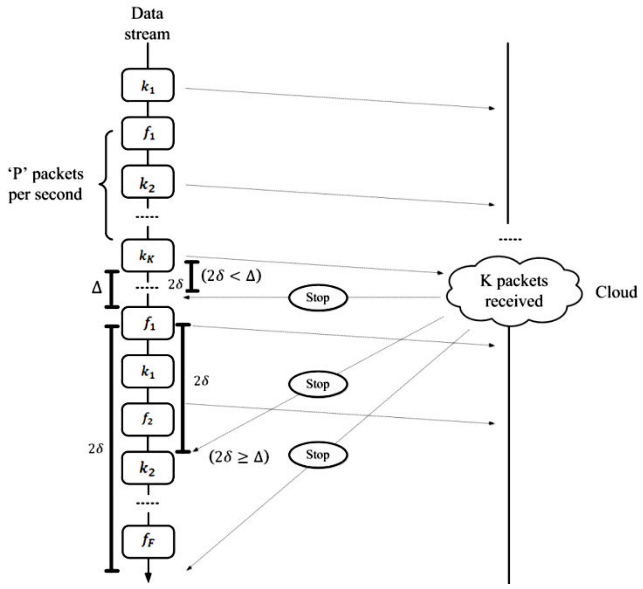
To predict the resulting bandwidth conservation, there is a need for statistically modeling the average number of packets transmitted per group ($\bar{T}$). After each group of 'K' original packets, the sender starts to compute the 'F' packets. The transmission of the FEC packets is interleaved with the commencement of transmission of a new original data packet group. Accordingly, there is a delay of Δ between the transmission time of the last original packet and first FEC packet of a group. Though, the minimum value of Δ is bounded by the calculation time of the FEC. There is a tradeoff involved while determining this parameter. By increasing theΔ, transmission of the superfluous packets is sent, when there is no loss of packet. In the case of packet loss, the client has to wait longer until sufficient FEC packets have received to start the decoding operation.

The average number of packets transmitted per group depends on the packet loss 'p', Δ and latency between the client and server δ. To recover all lost packets, the client must receive at least 'K' correct symbols

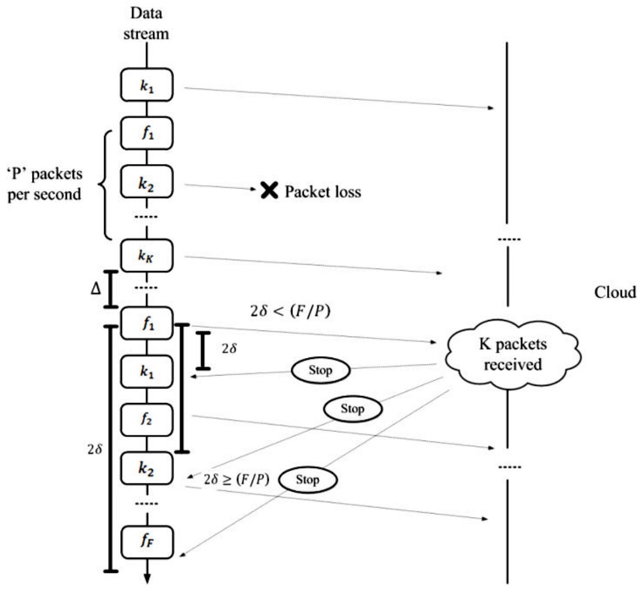
$$(1 - p)N \geq K \tag{1}$$

Since$N = K + f$, the average number of redundant packets to recover 'K' symbols

$$f = \frac{p}{1-p}K \tag{2}$$



(a)



(b)

Figure-3. Adaptive FEC system methodology: (a) No packet loss, K packets successfully received and (b) packet loss, at least one of K original packet lost.

This equation indicates the FEC packets will subject to the packet loss. But, due to the network latency, the feedback message will not reach immediately at the server that will transmit some additional packets. For modeling minimum overhead 'D', let us assume a constant packet size for the data streams and the total number of packets 'P' the server can send on the network per unit time. Packets contain original data or erasure correction information. To find expression for 'D', there is a need to differentiate two cases: none of 'K' original data packets are lost during the transmission and one of the original packets is lost. On a channel with random loss probability 'p', there is a possibility $(1-p)^K$ of correctly receiving the original packets by the client. The number of FEC packets that are transmitted while waiting for the feedback message is

$$D_{noloss} = \begin{cases} 0 & if\ 2\delta < \Delta \\ min(P(2\delta - \Delta), F) & if 2\delta \geq \Delta \end{cases} \quad (3)$$

For D_{noloss}, there is a need to distinguish two cases: the feedback message is received in time and $2\delta < \Delta$. In this case, no FEC packet will be transmitted. The feedback message is received later as $2\delta \geq \Delta$. Hence, the server will already have transmitted one FEC packet. For larger values of 2δ, the server should transmit all 'F' FEC packets before receiving the feedback message.

If at least one of the original 'K' packets is lost, the number of FEC packets that are transmitted is defined as, though it is not required by the client for retransmission,

$$D_{loss} = min(P2\delta, F) \quad (4)$$

D_{loss} is defined as the number of FEC packets that are transmitted but are not required for the reconstruction, where at least one of the original 'K' packets is lost. D_{loss} value depends on the round trip time of the network required to receive the feedback message from the client. Let us consider two cases: $2\delta < (F/P)$ and $2\delta \geq (F/P)$. In the first case, the feedback message arrives at the server before transmitting all FEC packets. In the second case, the feedback message arrives after all the FEC packets are transmitted. Packets transmitted before receiving feedback is counted as overhead

$$D = (1-p)^K D_{noloss} + (1-(1-p)^K) D_{loss} \quad (5)$$

The time to receive a feedback message always equal to 2δ. During this period, the server transmits $(2\delta . P)$ FEC packets. If the feedback message is lost, all the remaining packets are sent unnecessarily. If the feedback message is not lost, the average number of packets sent per group is given by

$$\bar{T}_{FBnotlost} = K + min\left(\frac{p}{1-p} \cdot K + D, F\right) \quad (6)$$

The feedback message is sent when the K packets of a specific group have been successfully received. Since the feedback message is sent over the same unreliable channel as the original data and FEC packets, it is subjected to the same loss probability 'p'. When the feedback message is lost, all the FEC packets will be transmitted. Therefore, $\bar{T}$ is expressed as

$$\bar{T} = \bar{T}_{FBnotlost} \cdot (1-p) + N \cdot p \quad (7)$$

As the consequence of losing the feedback message is high, it is evaluated where the client sends two feedback messages instantaneously after each other.

$$\bar{T} = \bar{T}_{FBnotlost} \cdot (1-p^2) + N \cdot p^2 \quad (8)$$

IA-FEC algorithm

```
Step 1: Initialization of user
Step 2: CSP activation
Step 3: User file upload
If ($file\ size\ \leq 0$)
Check the file
else
transfer the file to CSP
Step 4: Split the file into packets based on file size
Step 5: Convert data packet into binary digits
Step 6: Consider the file having maximum length //IA-FEC Check length of the string
For ($int\ i = 0; i \leq filecount; i++$)
Add extra zeros to the binary digits
Step 7: Set interleaving delay to each packet
Step 8: Transfer packet to the CSP
Step 9: Remove the extra zeros
Step 10: Check the error detection and correction //IA-FEC Check whether Adversary model is active
```

```
If Active
Packet loss is present
Error detection and correction using IA-FEC
Else
Error detection using IA-FEC
Reconstruct the packet
Merge the files and store at the server
Step 10: End
```

4. PERFORMANCE ANALYSIS

A. Test bed

In this section, the performance of the proposed IA-FEC system is evaluated. The proposed work is evaluated using the Netbeans IDE tool. The files are stored in OneDrive, a file hosting service operated by the Microsoft Corporation. OneDrive is used for storing the files and the personal data of the user in the cloud environment. Subsequently, the results for different packet loss configuration of 5, 10, 15 and 20% and different FEC group size of FEC-16, 32, 64 and 128 are evaluated. The group sizes are defined based on the number of original data packets K. This group size can be dynamically changed during a file transmission. Evaluation results for different static packet loss configurations of FEC-16, 32, 64 and 128 group sizes are presented. Parameter values for this evaluation section of proposed system are illustrated in Table-1.

Table-1. Parameters standard for the experimental analysis.

Parameter	Value
Iteration/ experiment	10
FEC group Size	16, 32, 64, 128
Loss type	Bursty
Average Packet loss (%)	5,10,15,20

B. Packet loss model

In this loss model, size of the loss interval is probabilistically estimated and the loss probability is set according to the Gilbert-Elliot model [24]. This model consists of two state, namely good or bad state with transition probabilities P_{BG} and P_{GB} respectively. In the good state configuration, all received packets are progressed ($P_G = 0$), while in the bad state, packets are randomly dropped according to a constructed packet loss value P_B. The resultant packet loss probability is defined as $p = (P_G P_{BG} + P_B P_{GB}) / (P_{GB} + P_{BG})$.

C. Performance evaluation

Figure-4 shows the difference between a traditional FEC code, layered interleaving and proposed method (IA-FEC) by plotting recovered latencies vs latency. Our test is conducted on a system with an Intel Core 2 processor running at 2.4 GHz, 2 GB RAM, and a 320 GB Serial Hard disk drive. The result denotes that the average of all experiments. The channel is organized to various packet loss of probability with different intervals. Table-2 shows the recovery latency analysis.

The latency of the proposed interleaving IA-FEC recovers the entire packet with minimum delay compared with existing method of FEC and layered interleaving method and reached its maximum whenever the burst reaches the peak value. The main goal of the proposed system is to minimize the redundancy rate to the level up to which the user just collects enough packets to reconstruct all corrupted or missing packets. Therefore, to make the best comparison, IA-FEC is compared with various group sizes in terms of a statistically configured redundancy rate (SRR). For 460 packets, the proposed IA-FEC requires minimum latency of about 76.67% and 41.67% than the FEC encoding and interleaving schemes. For 700 packets, the latency level of the proposed IA-FEC is minimum of about 89.13% and 50% than the FEC encoding and interleaving schemes.

Table-3 demonstrates the relative recovery rate achieved for bursty loss with single and feedback messages and with different measured packet losses of 5%, 10%, 15% and 20% and illustrated in Figure-5. For 15% packet loss, the average delay of the FEC-16 scheme is 275 ms, IA-FEC-16 scheme is 250 ms, FEC-64 scheme is 230 ms, IA-FEC-64 scheme is 210 ms, FEC-128 scheme is 170 ms and IA-FEC-128 scheme is 160 ms. The IA-FEC-16 scheme, IA-FEC-64 scheme and IA-FEC-128 scheme yields minimum average delay of about 9.09%, 8.69% and 5.88% than the FEC-16, FEC-64 and FEC-128 schemes.

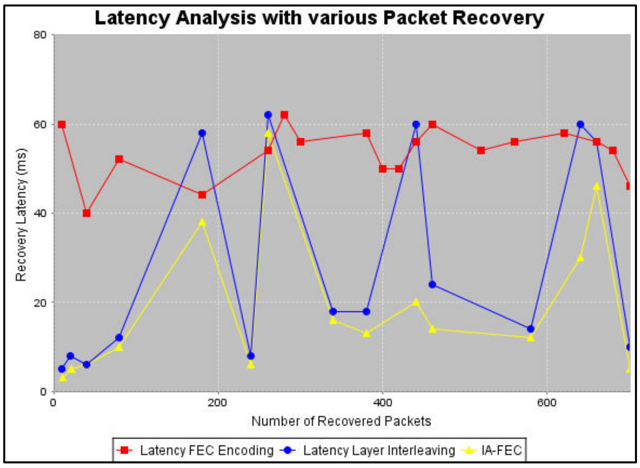


Figure-4. Latency analysis for various schemes.

Table-2. Recovery latency analysis.

Number of recovered packets	Recovery latency (ms)		
	Latency FEC encoding	Latency layer interleaving	IA-FEC
10	60	5	3
40	40	8	5
80	52	6	6
180	44	12	10

260	54	58	38
280	62	8	6
300	56	62	58
380	58	18	16
400	50	18	13
420	50	60	20
440	56	24	14
460	60	14	12
520	54	60	30
560	56	56	46
620	58	10	5
660	56	5	10
680	54	52	30
700	46	8	5

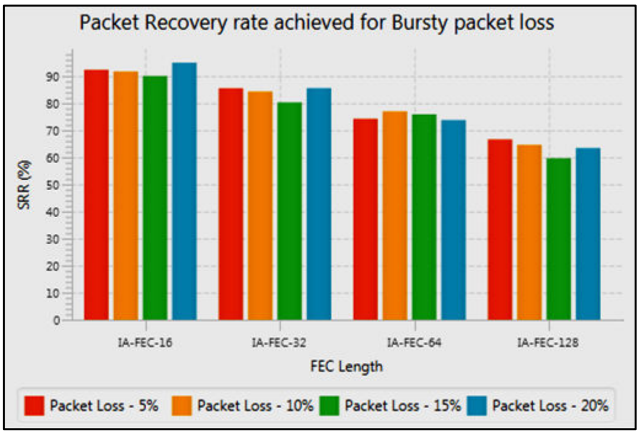


Figure-5. Packet recovery rate analysis for bursty packet loss.

Table-3. SRR analysis for various packet loss probabilities.

FEC length	Feedback	Packet Loss (%)			
		5	10	15	20
FEC-16	Single	92.35	85.54	74.26	66.66
FEC-32	Single	91.7	84.3	76.99	64.62
FEC-64	Single	90.08	80.3	75.92	59.68
FEC-128	Single	94.96	85.58	73.78	63.45

Table-4 shows the average delay analysis for various packet loss probabilities. Figure-6 presents the average delay of different IA-FEC system is compared with Traditional FEC. This comparison is evaluated in terms of various burst losses probability of 5%, 10% and 15%. From the graph, it is observed that the packet delay of IA-FEC is much lower than existing method.

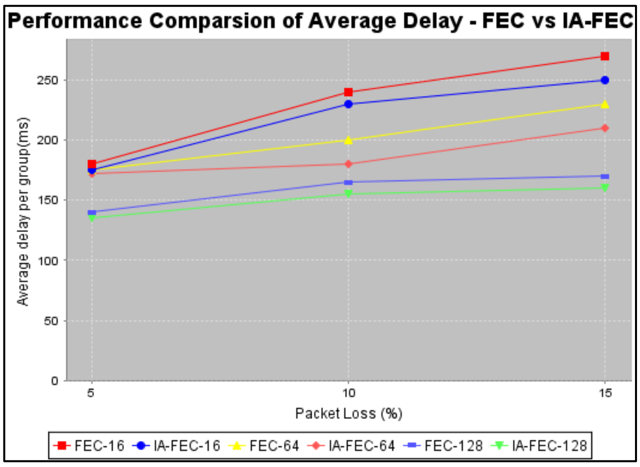


Figure-6. Average delay analysis for various packet loss percentages.

Table-4. Average delay analysis for various packet loss probabilities.

FEC length	Packet loss (%)		
	5	10	15
FEC-16	180	240	270
IAFEC-16	175	230	250
FEC-64	175	200	230
IAFEC-64	172	180	210
FEC-128	140	165	170
IAFEC-128	135	155	160

5. CONCLUSIONS

In this paper, an IA-FEC system is proposed. The cloud user interleaves original data packets with redundant padding packets. The cloud server detects lose packets and sends a feedback message when it has received enough redundant data packets to start the reconstruction process. In comparison with a statically configured redundancy rate, this feedback mechanism with the interleaving technique ensures that the user is able to reconstruct all packets under various packet loss environments. The system has been widely evaluated for bursty loss pattern with various loss probabilities. By security and performance analysis, the proposed method having minimum average delay and maximum recovery rate for all configured system of IA-FEC-16, IA-FEC-32, IA-FEC-64 and IA-FEC-128 is described. The proposed scheme is highly efficient in recovering the singleton losses almost quickly and recovers from bursty data losses is demonstrated.

REFERENCES

[1] E. Amazon. 2015. Amazon web services. Available in: http://aws. amazon. com/es/ec2/(November 2012).

[2] N. Gohring. 2008. Amazon's S3 down for several hours. Online at http://www. pcworld. com/businesscenter/article/142549/amazons s3 down for several hours. Html.

[3] C. Wang, Q. Wang, K. Ren, N. Cao and W. Lou. 2012. Toward secure and dependable storage services in cloud computing. IEEE transactions on Services Computing. 5: 220-232.

[4] D. L. Gazzoni Filho and P. S. L. M. Barreto. 2006. Demonstrating data possession and uncheatable data transfer. IACR Cryptology ePrint Archive. 2006: 150.

[5] M. A. Shah, R. Swaminathan and M. Baker. 2008. Privacy-Preserving Audit and Extraction of Digital Contents. IACR Cryptology ePrint Archive. 2008: 186.

[6] K. D. Bowers, A. Juels and A. Oprea. 2009. Proofs of retrievability: Theory and implementation. in Proceedings of the 2009 ACM workshop on Cloud computing security. pp. 43-54.

[7] D. Fountain. Why raptor is better than reed-solomon for streaming applications [Online]. Available: http://www.qualcomm.com/media/ documents/why-raptor-codes-are-betterreed-solomon-codes-streamingapplications.pdf

[8] L. Libman and A. Orda. 2006. Optimal packet-level FEC strategies in connections with large delay-bandwidth products. IEEE transactions on wireless communications. 5: 1645-1650.

[9] F. A. Ali, P. Simoens, W. Van de Meerssche and B. Dhoedt. 2014. Bandwidth efficient adaptive forward error correction mechanism with feedback channel. Journal of communications and networks. 16: 322-334.

[10] M. K. Kadiyala, R. Pendse and K. R. Namuduri. 2010. On the dependence of burst losses on the packet inter-arrival times in VoIP. in GLOBECOM Workshops (GC Wkshps), 2010 IEEE. pp. 975-979.

[11] M. K. Kadiyala, R. Pendse and a. K. Namuduri. 2011. On the impact of the packet inter arrival times on burst loss patterns in VoIP. Journal of Inter Services and Applications, Jan.

[12] Q. Wang, C. Wang, K. Ren, W. Lou and J. Li. 2011. Enabling public auditability and data dynamics for storage security in cloud computing. IEEE transactions on parallel and distributed systems. 22: 847-859.

[13] M. A. Shah, M. Baker, J. C. Mogul, and R. Swaminathan. 2007. Auditing to Keep Online Storage Services Honest. in HotOS.

[14] H. Shacham and B. Waters. 2013. Compact proofs of retrievability. Journal of cryptology. 26: 442-483.

[15] Y. Sohn, J. Hwang and S.-S. Kang. 2012. Adaptive packet-level FEC algorithm for improving the video quality over IEEE 802.11 networks. International Journal of Software Engineering and Its Applications. 6: 27-34.

[16] M.-F. Tsai, C.-H. Ke, H.-M. Liang and H.-Y. Huang. 2011. Forward error correction with interleaving mechanism combining cognitive Technology for video streaming over wireless networks. in Wireless and Pervasive Computing (ISWPC), 2011 6th International Symposium on. pp. 1-6.

[17] C. Yang, C. Liu, X. Zhang, S. Nepal and J. Chen. 2015. A time efficient approach for detecting errors in big sensor data on cloud. IEEE Transactions on Parallel and Distributed Systems. 26: 329-339.

[18] C. Barakat and E. Altman. 2002. Bandwidth tradeoff between TCP and link-level FEC. Computer networks. 39: 133-150.

[19] J.-C. Bolot, S. Fosse-Parisis and D. Towsley. 1999. Adaptive FEC-based error control for Internet telephony. in INFOCOM'99. Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE. pp. 1453-1460.

[20] P. Frossard. 2001. FEC performance in multimedia streaming. IEEE Communications Letters. 5: 122-124.

[21] J. Cai and C. W. Chen. 2001. FEC-based video streaming over packet loss networks with pre-interleaving. in Information Technology: Coding and Computing, 2001. Proceedings. International Conference on. pp. 10-14.

[22] M. Claypool and Y. Zhu. 2003. Using interleaving to ameliorate the effects of packet loss in a video stream. in Distributed Computing Systems Workshops, 2003. Proceedings. 23rd International Conference on. pp. 508-513.

[23] W.-T. Tan and A. Zakhor. 1999. Error control for video multicast using hierarchical FEC. in Image Processing, 1999. ICIP 99. Proceedings. 1999 International Conference on. pp. 401-405.

[24] J.-P. Ebert and A. Willig. 1999. A Gilbert-Elliot bit error model and the efficient use in packet level simulation.