An Adaptive Weighted Online AL-FEC Algorithm over Mobile Multicast Networks

Christos Bouras, Nikolaos Kanakis
Computer Technology Institute and Press “Diophantus”, Patras, Greece
Computer Engineering and Informatics Department, University of Patras, Greece
bouras@cti.gr, kanakisn@cti.gr

***Abstract*—A crucial point on the delivery of multicast content over mobile multicast networks is the utilization of Forward Error Correction (FEC) codes on the application layer (AL-FEC) so as to introduce robustness against arbitrary erasures on different recipients without the need of inefficient common error control methods based on data retransmission. FEC is a feedback free error recovery method where the sender introduces redundant data in advance with the source data enabling the recipients to recover from different arbitrary packet losses. The main issue on the efficient application of AL-FEC protection is the adaptation of the introduced redundancy to the current network conditions, in order to avoid network resources wastage ensuring at the same time sufficient transmission robustness. Based on this, in this work we present an adaptive weighted online algorithm aiming at the efficient application of RaptorQ AL-FEC codes over mobile multicast services. The proposed algorithm adapts the introduced AL-FEC transmission overhead exploiting at first the reception reports ability defined by several mobile multicast standards and thereafter the performance properties of the newly introduced RaptorQ FEC codes. We introduce the competitive framework of the efficient application of AL-FEC protection over mobile multicast networks, in the context of which, we design and analyze the adaptive online weighted algorithm and we further provide a series of simulation results to analyze the performance of the proposed scheme.**

***Keywords*-forward error correction, RaptorQ codes, multicast delivery, adaptive algorithms, online algorithms**

I. Introduction

Forward error correction (FEC) is a method for error control of data transmission adopted in several mobile multicast standards. In multicast delivery, the FEC encoding significantly reduces the effect of independent losses at different receivers, while achieving a reduction in the rate of packet loss according to the introduced redundancy by the FEC encoder, resulting in large mitigation to the costly need of lost packets retransmission. Based on the above, several mobile multicast standards [1], [2] recommend the use of FEC on application layer, and more specifically, Raptor codes family [3] are adopted due to their high performance. However, FEC protection comes with its own cost since controlling the introduced redundancy is not a trivial issue. The multicast sender should decide on the redundancy will introduce to the transmission so as to ensure that the multicast recipients will be able to recover independent data losses while, at the same time the redundant information should be adapted to the current reception conditions to avoid resources wastage. Based on this, the efficient application of AL-FEC protection can be achieved by a multicast transmitter enabled to adapt the introduced AL-FEC redundancy according to the current reception conditions. The design of an algorithm adapting the introduced AL-FEC transmission overhead can be reduced in the basis of an online problem [4].

In general, online algorithms [5] are used to confront problems where the input of the algorithm is not available in advance. Subsequently, online algorithms have to generate output without knowledge of the entire input since input information arrives in the future and is not accessible at present. The effectiveness of online algorithms is evaluated using competitive analysis. The main concept of competitiveness is to compare the output generated by an online algorithm to the output produced by an optimal offline algorithm which knows the entire request sequence in advance and can serve it with minimum cost.

Online algorithms are utilized in many research fields of mobile networks as the work presented in [6], where the frequency assignment problem is examined through distributed online algorithms. In the context of energy constraints and the design of routing algorithms, the authors of [7] propose an online algorithm on maximizing the throughput of multihop radio networks. Moreover, in [8] online algorithms are utilized on multicast routing problems over energy-constrained adhoc networks. Finally, the work presented in [9] introduces a competitive online algorithm in terms of energy efficiency and delay in scheduling problems over wireless multicast environments. The work presented in [10] proposes a data selection policy where, in the concept of competitive analysis, the decision of transmitting source data, retransmitting a packet or transmitting a redundant codeword is investigated. Finally, the authors of this paper introduced in [11] an online framework for the utilization of online algorithms on the efficient application of AL-FEC protection problem over mobile multicast networks evaluating the first attempt of a naive randomized online algorithm for the stated AL-FEC policy online problem. Moreover, the same authors presented in [12] a deterministic online algorithm based on weights assignment in each AL-FEC processed packet adapting the introduced AL-FEC overhead according to some encoding properties of RaptorQ AL-FEC code.

In this work, we introduce an enhanced variation of the weighted online algorithm presented in [12]. We aim to enhance its protection efficiency and introduce adaptation nature according to the network reception conditions exploiting the multicast clients ability to determine the number of lost AL-FEC symbols. More precisely, the proposed algorithm extends the AL-FEC weighted online algorithm introducing the ability to the multicast sender to monitor previous outcomes of the transmitted AL-FEC symbols to multicast UEs and utilizing this information can adapt the introduced AL-FEC transmission overhead accordingly. To this direction, we provide an analytical competitive network model, defining the optimal offline adversary and thereafter we propose an adaptive online AL-FEC algorithm.

The rest of this paper is organized as follows: In Section II we describe the competitive framework introducing the network model we utilize and we further describe the optimal offline algorithm. In Section III we present the proposed adapted online AL-FEC algorithm followed by an analysis on its operation. In Section IV we analyze the performance of the proposed scheme providing several simulation results against the optimal offline instance. Finally, in Section V we conclude the presented work and we draw some possible future steps that can follow this work.

II. Competitive Framework

In this section we describe and analyze the online problem we are dealing with and we present the utilized network model under which we design the proposed online algorithm. Moreover, we provide a brief presentation of the RaptorQ code, the AL-FEC scheme we utilize for the proposed adaptive online algorithm, and we further describe the optimal offline algorithm and we analyze its performance.

A. Network Model

Defining the online AL-FEC application problem, the multicast source takes as input a sequence of packets and a set of encoding parameters and has to produce as output AL-FEC encoded symbols deciding on the amount of AL-FEC overhead that will be introduced to the transmission. The network model we introduce refers to a typical mobile multicast transmission environment, where the same data are transmitted to a fraction of users participating in the multicast delivery through a shared unreliable radio channel.

In this work we utilize RaptorQ codes as the AL-FEC encoding scheme. RaptorQ FEC is the newest member of Raptor codes family providing powerful capabilities on the AL-FEC protection application. RaptorQ code is a fountain code, meaning that as many encoding symbols as desired can be generated by the encoder on-the-fly from the source symbols of a source block of data. The encoding process is systematic since *n* encoding symbols are produced from *k*<*n* source symbols, so as the original source symbols are within the stream of the transmitted symbols. RaptorQ can encode up to 56403 source symbols into a source block and can generate up to 16777216 encoding symbols from the source symbols providing the ability to deliver files up to 3.4 GB as a single source block. The encoding process of such a FEC code provides the ability to the decoder to recover the whole source block from any set of encoding symbols only slightly more in number than the source symbols. More precisely, the performance of an AL-FEC code can be described by the decoding failure probability of the code, denoting the probability the RaptorQ decoder to fail on successfully reconstructing the protected data as a function of the source block size and the number of received symbols. The decoding failure probability of RaptorQ code can be modeled by (1) [13]:

$$p_{f_{RQ}}(n,k) = \begin{cases} 1, & \text{if } n < k \\ 0.01 \times 0.01^{n-k}, & \text{if } n \geq k \end{cases} \quad (1)$$

In (1), $p_{f_{RQ}}(n,k)$ denotes the probability of a failed decode of a RaptorQ protected block with k source symbols if n encoding symbols have been received.

In this work, we assume the transmission of a packet sequence with independent packet loss masks applied to each multicast receiver according to an examined packet loss rate. In each packet sequence, each packet is denoted by the triplet $\{uid, sbn, r_{il}\}$ where:

- uid: is a unique ID identifying each AL-FEC resulting packet
- sbn: is the number of the FEC source block the examined packet is organized to
- r_{il}: defines if the examined packet was not received by the receiver i with the boolean l set to 0 if packet was not received

The behavior of the network is modeled as a loss transcript, consisting of the values of the boolean variables r_{il}. In more detail, in the general multicast network model we consider, the values r_{il} may be set arbitrarily, allowing for bursty periods of loss which need not to be correlated across the multicast receivers. More precisely, the packet loss pattern applied to the sequence of transmitted packets is denoted by p, which is the average network packet loss rate taking values in the range $[0,1]$. At each multicast receiver, a packet loss mask is applied independently based on the value of p. The packet erasures are randomly distributed at the receivers as illustrated in Fig. 1, where an instance of the successful or not reception of 10 transmitted packets at 5 multicast receivers is presented with the value of p fixed at 0.2.

User 1	(1,0)	(1,1)	(1,1)	(1,1)	(1,1)	(1,1)	(1,1)	(1,1)	(1,0)	(1,0)
User 2	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)	(2,1)
User 3	(3,1)	(3,0)	(3,1)	(3,1)	(3,0)	(3,0)	(3,0)	(3,1)	(3,1)	(3,0)
User 4	(4,1)	(4,1)	(4,1)	(4,1)	(4,1)	(4,1)	(4,0)	(4,1)	(4,1)	(4,1)
User 5	(5,1)	(5,1)	(5,1)	(5,0)	(5,1)	(5,1)	(5,1)	(5,1)	(5,1)	(5,1)

Fig. 1: An Instance of Packet Erasures

B. Optimal Offline Algorithm

Considering the cost of the optimal offline algorithm, we assume that a multicast transmitter, with a priori knowledge of the packet losses pattern of the network, will introduce a certain number of redundant symbols. Since the AL-FEC policy problem is a cost minimization problem, the optimal policy from the transmission cost perspective on the selection of the AL-FEC overhead which a multicast source should introduce to the transmission is not the introduction of a huge amount of overhead aiming to enable the higher packet loss user to successfully recover the transmitted object. Subsequently, the optimal algorithm will introduce the minimum number of required redundant symbols so as to cope with the average value of packet loss aiming to satisfy as many users as possible.

The scheme that can ensure the optimal selection of the transmission overhead is described by a multicast source that selects the introduced redundancy to a value close to the average packet loss rate of the network as defined in [14] given the recovery properties of the utilized AL-FEC code. In the present analysis the multicast sender can exploit the exceptional recovery properties of RaptorQ code. RaptorQ provides a practically zero reception overhead since, as described in (1), can achieve the specified threshold of the decoding failure probability requiring to receive no more additional encoding symbols than the number of the transmitted source symbols. Subsequently, the optimal AL-FEC selection policy can introduce as many repair symbols as the average number of lost symbols in the multicast users. Based on this, the number of repair symbols r the optimal offline algorithm will introduce in each source block of size sbl symbols is calculated as follows: $r = (sbl + r) \cdot p$. Consequently, the cost of the optimal AL-FEC policy algorithm can be computed as: $OPT = sbl + r$.

III. Adaptive Weighted Online Algorithm

As already mentioned, the proposed online algorithm extends the online scheme presented in [12] and comes to enhance its performance, introducing an adaptive variation based on the outcome of previous multicast deliveries of the transmitted object. For the purposes of the presented algorithm, we suppose that the multicast source can monitor and log the outcome of each multicast delivery. To clarify this assumption, we refer that several mobile multicast standards define a post-delivery procedure to provide extra features (e.g. file repair capabilities) for the multicast download delivery. Based on this, a multicast UE is able to determine, for each source block of each file, which source symbols should have been received but have not and is also able to determine the number of symbols it has received. Therefore, a multicast sender is able to determine which UEs failed to decode the AL-FEC protected object.

On this basis, the proposed adaptive algorithm computes a quantity called $factor$ upon the completion of a multicast transmission. The quantity $factor$ denotes the outcome of the monitored multicast delivery, in terms of UES coverage. With

Algorithm 1 Adaptive Weighted AL-FEC Algorithm

procedure $(symbols, sbl, targetThreshold)$
    $compute\ factor$
    if $factor \leq targetThreshold$ **then**
        $threshold \leftarrow threshold - (0.05 * threshold)$
    else
        $threshold \leftarrow threshold + (0.05 * (1 - threshold))$
    end if
    $count \leftarrow 0$
    for all symbols **do**
        $symbol.w \leftarrow log_2(symbol.uid)/log_2(sbl)$
        if $symbol.w \leq threshold$ **then**
            $count \leftarrow count + 1$
        end if
        if $symbol.uid \bmod sbl = 0$ **then**
            transmission overhead $\leftarrow count/sbl$
        end if
    end for
end procedure

the term UEs coverage we denote the fraction of UEs that were able to successfully reconstruct the AL-FEC protected data compared to the fraction of UEs participating in the multicast delivery. Hence, $factor$ is computed as follows:

$$factor \leftarrow \frac{\#decoded\ UEs}{\#UEs}$$

where a $decoded\ UE$ is a multicast client who successfully decoded the AL-FEC protected object.

In more detail, the adaptive algorithm takes as input a sequence of symbols, assuming one symbol per packet, the length of the source blocks that will be produced and a quantity called $targetThreshold$. The value of $targetThreshold$ determines the UEs coverage that the algorithm should achieve. Further to that, $targetThreshold$ initializes at the first transmission round the value of the $threshold$ that the weighted algorithm of [12] utilizes. Furthermore, in each AL-FEC symbol is assigned a quantity weight w, with the value of this quantity, in conjunction with the value of the $threshold$, determining if the processed symbol will be included in the computation of the introduced AL-FEC transmission overhead.

The proposed adaptive Algorithm 1 is designed to operate in transmission rounds and, based on the values of the computed quantities described above, the algorithm operates as follows: For each transmission round the algorithm computes the value of the $factor$ of the previous round. Thereafter, the algorithm adjusts the threshold t leveraging its knowledge on the network's state in previous delivery rounds. In the case that the computed $factor$ is less or equal to the value of the requested by the user $targetThreshold$, the algorithm reduces the $threshold$ value utilized by the weighted algorithm of [12] in order to increase the number of introduced AL-FEC repair symbols compared to the number of repair symbols introduced to the previous round and, therefore the algorithm is able to enhance the introduced AL-FEC protection robustness. On the other hand, when the value of $factor$ overcomes the requested $targetThreshold$, the algorithm reduces the number of introduced AL-FEC redundancy by increasing the value of $threshold$ in order to avoid resources wastage with respect to the requested AL-FEC protection performance. Finally after the parameter's adaptation phase, the online algorithm applies the algorithm of [12], processing the sequence of packets to be AL-FEC protected where decides, according to the computed values of each symbol's weight w and $threshold$, if the processed symbol will contribute in the introduced AL-FEC redundancy i.e., if a repair AL-FEC symbol will be produced for the current AL-FEC source symbol.

Regarding the cost of the proposed adaptive online algorithm, for each transmission round it can be computed by [12], as $ALG = sbl + sbl^{threshold}$. However, the presented online algorithm adapts the value of $threshold$ for each round of multicast transmission and does not utilize a constant value, given by the user, as the online algorithm of [12] does.

Comparing the proposed adaptive Algorithm 1, the weighted algorithm of [12] utilizes just the assigned weight w of each AL-FEC symbol, computed according to the uid of the symbol and the encoding property of sbl, to decide if the specified AL-FEC symbol will participate in the AL-FEC transmission overhead according to the hardcoded value of $threshold$. On the other hand, the proposed adaptive online algorithm monitors previous outcomes of the transmitted AL-FEC encoding symbols to the multicast participants and through the utilization of this information is able to adapt the value of $threshold$ in order to achieve the performance denoted by the $targetThreshold$, which is requested by the user.

IV. Performance Evaluation

In this section we provide simulation results evaluating the achieved performance of the proposed online AL-FEC adaptation algorithm. At first, we present simulation results describing the operation concept of the proposed algorithm, and thereafter we provide results comparing the impacts of the proposed algorithm on several performance perspectives against the performance of the optimal offline algorithm.

A. Algorithm Convergence

In this paragraph we illustrate the operation concept of the proposed adaptive weighted online algorithm. We investigate how the proposed adaptive algorithm converges in a stable introduced AL-FEC redundancy state, since the algorithm operates in transmission rounds. More precisely, Fig. 2 and Fig. 3 present how our online scheme adapts the introduced AL-FEC transmission overhead for 100 consecutive rounds of multicast transmission simulation with the packet loss rate fixed at 5% and 15% respectively. The provided results consider the transmission of a source block of size 256 symbols at 100 multicast UEs evaluating different initial values for the algorithm target threshold.

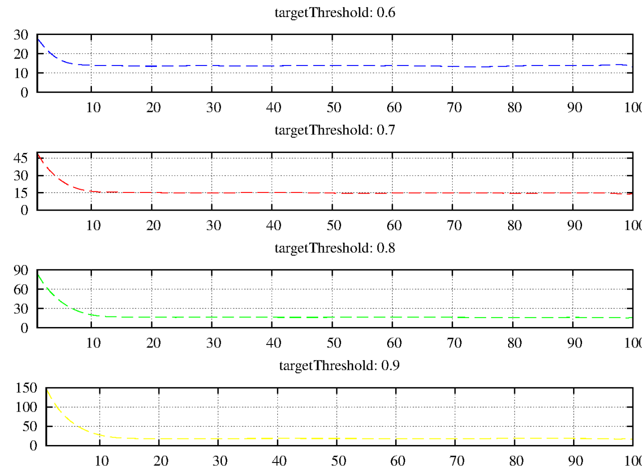

Fig. 2: Overhead Symbols vs. Transmission Round
PLR: 5%

Investigating the curves presented in Fig. 2 for four different values of the parameter $targetThreshold$, we can immediately notice that the adaptive algorithm reaches a stable state, i.e. a near to constant amount of introduced AL-FEC redundant symbols after almost ten consecutive rounds of multicast delivery simulation. In more detail, in the first case, where the $targetThreshold$ value is set to 0.6 converges too early, almost after five transmission rounds. This behavior is due to the match of the simulated network's packet loss rate, set to 5%, with the evaluated value of the $targetThreshold$ value. To clarify the latter, according to the requested $targetThreshold$, the algorithm introduced initially an amount of overhead symbols close to the number of symbols required to achieve the requested protection with respect to the simulated packet loss rate of the network. Hence, the readily convergence of the adaptive algorithm to a stable condition is expected. In the second case, where the $targetThreshold$ is set to 0.7 we can remark that the adaptation requires more transmission rounds to achieve a stable state compared to the previous case, since the algorithm introduces an almost constant number of repair AL-FEC symbols after eight transmission rounds. This behavior is due to the higher value selected for the $targetThreshold$, since the higher initial threshold, according to the requested $targetThreshold$, indicates higher number of initial repair symbols with the number of repair symbols, required to achieve the $targetThreshold$ performance be slightly more compared to the previous case. In the case of $targetThreshold$ fixed at 0.8 we notice again the same behavior, meaning that the algorithm requires more rounds to converge compared to the second case. Once again, this behavior is a direct consequence of the higher requested value of $targetThreshold$. The same lies for the last case, simulated in Fig. 2, where the $targetThreshold$ is fixed to the highest value, at 0.9.

Regarding the simulation results presented in Fig. 3, we examine the behavior of the proposed adaptive weighted online algorithm for the same $targetThreshold$ values as

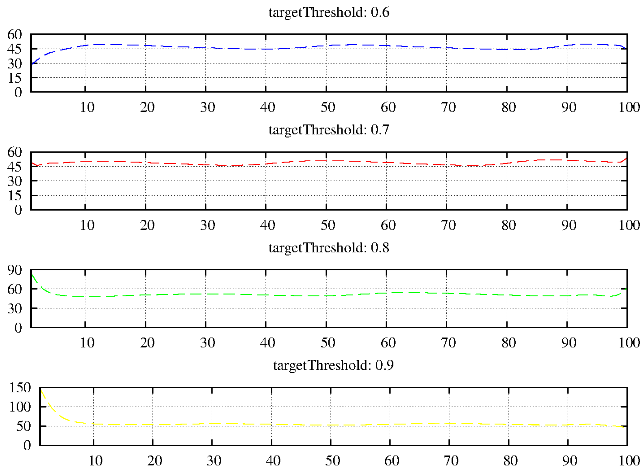

Fig. 3: Overhead Symbols vs. Transmission Round PLR: 15%

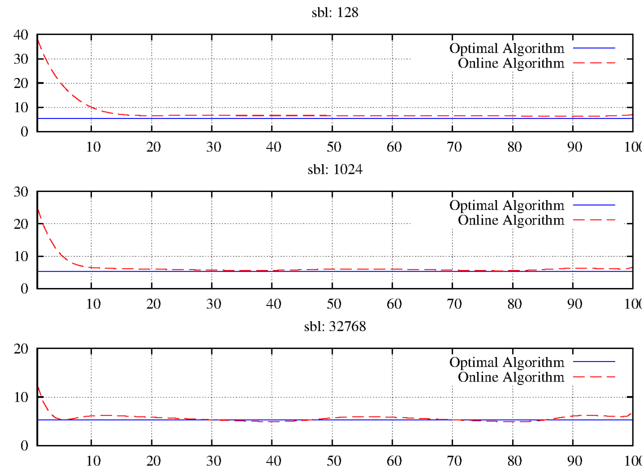

Fig. 4: AL-FEC Transmission Overhead(%) vs. Transmission Round

in Fig. 2 but for a higher evaluated value of packet loss rate, fixed at 15%. In general, the plotted curves verify once again that the adaptive algorithm converges in a stable state after about ten transmission rounds. Analyzing each individual case of $targetThreshold$ value, we can remark that in the first case where the $targetThreshold$ is fixed at 0.6, the adaptive algorithm increases the introduced redundancy since the $targetThreshold$, that also defines the initial threshold of the online algorithm, is selected to a low value regarding the packet loss rate has to confront with. In the second case, the plotted curve reveal an almost perfect match between the initial threshold and the packet loss rate due to the increase of the $targetThreshold$ to 0.7. For the last two cases, where the $targetThreshold$ is fixed at 0.8 and 0.9 respectively, we can denote that since the $targetThreshold$ value is increased and therefore the initial threshold becomes more and more higher compared to the "ideal" value of the second case, the algorithm's convergence becomes more and more slow.

B. Introduced AL-FEC Transmission Overhead

In this part of the provided performance evaluation we provide simulation results comparing the proposed adaptive weighted online algorithm with the optimal offline algorithm. More precisely, in Fig. 4 we compare the introduced AL-FEC transmission overhead, in terms of percentage, for 100 consecutive rounds of multicast transmission simulation with the packet loss rate fixed at 5% simulating the transmission of a source block of size 128, 1024 and 32768 symbols at 100 multicast UEs with the $targetThreshold$ value fixed at 0.8.

Regarding the results presented in Fig. 4, we can immediately remark the impacts of combining the adaptation feature of the proposed online algorithm with the weighted online algorithm of [12], since the presented evaluation results refer to different values of sbl. In more detail, all of the three cases of different sbl verify once again that the adaptive algorithm converges at worst after ten transmission rounds, achieving an almost constant amount of introduced AL-FEC transmission overhead and reaching the performance of the optimal offline algorithm. Moreover, we can notice that as the sbl increases the rate of the algorithm's convergence increases too. This fact is a direct consequence of both the initial threshold the adaptive algorithm utilizes, as well as the performance properties of the RaptorQ AL-FEC code. At this point, we have to notice that RaptorQ is able to achieve enhanced recovery performance as the sbl increase [14]. Further to the previous remarks, we can observe that for the highest size of sbl i.e., 32768 symbols, the variation of the introduced transmission overhead between the stabilized value is more pronounced compared to the lower sbl cases. This is something anticipated for large AL-FEC source blocks, and this is the reason why the algorithm converges faster in the last simulated case of Fig. 4.

C. UEs Coverage

In the last part of the provided simulation results, we illustrate the performance of the proposed adaptive algorithm on the UEs coverage. In Fig. 5 we present how the achieved UEs coverage varies during consecutive transmission rounds with respect to the value of the AL-FEC transmission overhead, the adaptive algorithm introduces on each transmission round. For this evaluation, we simulate 100 multicast transmission rounds over 100 UEs with the average packet loss rate is fixed at 5% and, the sbl and the $targetThreshold$ is fixed at 1024 and 0.7 respectively.

Observing the achieved UEs coverage values plotted in Fig. 5, we can remark that for the first ten consecutive transmission rounds the adaptive algorithm constantly reduces the introduced overhead and hence the achieved coverage, until it reaches the value of the requested $targetThreshold$. Indeed, we observe that when the UEs coverage reaches the 70% the following values are scattered around this value. In fact, this is the operational concept of the introduced adaptive algorithm, since it adapts the amount of introduced transmission overhead until the achieved UEs coverage matches the requested value

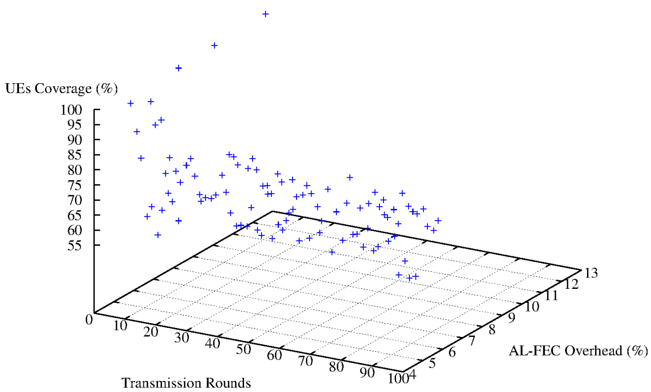

Fig. 5: UEs Coverage vs. Transmission Round & AL-FEC Transmission Overhead(%)

of $targetThreshold$. Hence, in the simulated case where the initial amount of introduced overhead is high for the coverage it has to achieve with respect to the network's packet loss rate, the adaptive algorithm gradually reduces the introduced overhead until it reaches the requested UEs coverage value i.e., 70%.

V. Conclusions & Future Work

In this work, we have presented an adaptive weighted online algorithm on the online AL-FEC application problem. The proposed adaptive online scheme extends the weighted online algorithm presented in [12] and addresses its shortcomings. Since the major requirement on achieving an efficient application of AL-FEC protection is the introduction of enough redundancy that can ensure robustness on the multicast transmission while, at the same time can achieve efficient utilization of the network's resources, we have introduced an adaptive scheme that utilizes the weight assignment mechanism of the algorithm of [12] and enforces it by introducing an adaptive mechanism on the current network's packet loss conditions exploiting monitoring capabilities of mobile multicast standards.

We have provided the analysis of the network model we have utilized for the design and evaluation of the proposed AL-FEC adaptive algorithm and we have analyzed the optimal offline algorithm for the stated online problem. Thereafter, we have introduced the proposed online adaptive AL-FEC algorithm followed by a thorough analysis of its operational concept and we have provided extensive simulation results that investigate the performance of the proposed adaptive scheme evaluating its efficiency under several reception conditions and AL-FEC encoding parameters. From the simulation results we have presented, we were able to verify that the adaptive algorithm can achieve a fast convergence to a stable state, regarding the introduced AL-FEC transmission overhead, over several different reception conditions and algorithm's parameters. Moreover, we have verified that the adaptive algorithm provides a significant gain on the introduced AL-FEC redundancy, providing efficient resource utilization, and achieving at the same the requested goals for the UEs coverage at a very short time of adaptation. Finally, we were able to notice that the adaptive algorithm provides more smooth adaptation when the transmitted object is partitioned into small AL-FEC source blocks while, on the other hand for high values of the source block size the algorithm achieves faster convergence.

Regarding some future steps that can follow this work, we could design several online schemes over different perspectives since the online AL-FEC application problem is a newly introduced online problem. Furthermore, we could investigate the impacts of online algorithms over the application of AL-FEC protection over unicast environments, since it is our belief that an online scheme in conjunction with the powerful RaptorQ AL-FEC code can totally replace common methods of protection against data losses.

References

[1] 3GPP, "Multimedia Broadcast/Multicast Service (MBMS); Protocols and codecs (Release 10)," 3rd Generation Partnership Project (3GPP), TS 26.346, 2011. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/26346.htm

[2] ETSI, "Digital Video Broadcasting (DVB); Guidelines for the implementation of DVB-IPTV Phase 1 specifications; Part 3: Error Recovery; Sub-part 2: Application Layer - Forward Error Correction (AL-FEC)," ETSI, TS 102 542-3-2, 2011.

[3] M. Watson, T. Stockhammer, and M. Luby, "Raptor Forward Error Correction (FEC) Schemes for FECFRAME," RFC 6681, Internet Engineering Task Force, Aug. 2012. [Online]. Available: http://www.ietf.org/rfc/rfc6681.txt

[4] R. Motwani and P. Raghavan, "Algorithms and theory of computation handbook," M. J. Atallah and M. Blanton, Eds. Chapman & Hall/CRC, 2010, ch. Randomized algorithms. [Online]. Available: http://dl.acm.org/citation.cfm?id=1882757.1882769

[5] A. Borodin and R. El-Yaniv, *Online computation and competitive analysis*. New York, NY, USA: Cambridge University Press, 1998.

[6] J. Janssen, D. Krizanc, L. Narayanan, and S. Shende, "Distributed Online Frequency Assignment in Cellular Networks," *Journal of Algorithms*, vol. 36, no. 2, pp. 119 – 151, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0196677499910684

[7] L. Lin, N. Shroff, and R. Srikant, "Asymptotically optimal energy-aware routing for multihop wireless networks with renewable energy sources," *Networking, IEEE/ACM Transactions on*, vol. 15, no. 5, pp. 1021 –1034, oct. 2007.

[8] W. Liang and X. Quo, "Online multicasting for network capacity maximization in energy-constrained ad hoc networks," *Mobile Computing, IEEE Transactions on*, vol. 5, no. 9, pp. 1215 –1227, sept. 2006.

[9] A. El Gamal, C. Nair, B. Prabhakar, E. Uysal-Biyikoglu, and S. Zahedi, "Energy-efficient scheduling of packet transmissions over wireless networks," in *INFOCOM 2002. Twenty-First Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings. IEEE*, vol. 3, 2002, pp. 1773 – 1782 vol.3.

[10] Y. Bartal, J. Byers, M. Luby, and D. Raz, "Feedback-free multicast prefix protocols," in *Computers and Communications, 1998. ISCC '98. Proceedings. Third IEEE Symposium on*, jun-2 jul 1998, pp. 135–141.

[11] C. Bouras, N. Kanakis, V. Kokkinos, and A. Papazois, "Deploying AL-FEC with Online Algorithms," in *Next Generation Mobile Apps, Services and Technologies (NGMAST), 2013 Seventh International Conference on*. IEEE, 2013, pp. 175–180.

[12] C. Bouras and N. Kanakis, "A Competitive AL-FEC Framework over Mobile Multicast Delivery," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, jul. 2013.

[13] 3GPP, "Rationale for MBMS AL-FEC Enhancements," 3rd Generation Partnership Project (3GPP), Tdoc S4-110449, 2011.

[14] C. Bouras, N. Kanakis, V. Kokkinos, and A. Papazois, "Application layer forward error correction for multicast streaming over LTE networks," *International Journal of Communication Systems*, 2012. [Online]. Available: http://dx.doi.org/10.1002/dac.2321