Towards integrating Quantize-Map-Forward relaying into LTE

Emre Atsan
EPFL, Switzerland

Raymond Knopp
Institute Eurècom, France

Suhas Diggavi
UCLA, USA

Christina Fragouli
EPFL, Switzerland

***Abstract*—We present a method to integrate the Quantize-Map-Forward (QMF) relaying scheme [1] into the standard LTE operation, for a two-relay diamond network configuration. Our approach implements QMF using mainly existing LTE modules and functionalities, and results in minimal changes in the standard link-layer LTE operation. In particular, the destination operation is only affected in that we adapt the log-likelihood ratio (LLR) calculations at the decoder input to take into account the existence of relays; thus, the decoding complexity and operations (apart the LLR calculations) are not modified. We report extensive performance evaluations of our scheme using the OpenAirInterface (OAI) link-level simulation tools.**[1]

I. Introduction

One of the main novelties LTE-A (Long Term Evolution - Advanced) will bring on LTE is the addition of relay nodes to the network architecture. Existing LTE systems use direct communication links between the User Equipment (UE) and the basetation (eNB). LTE-A relaying will enable information flow through relays between the UE and eNB, with the goal of extending the coverage around the cell edges and increasing the capacity in hotspots. The diamond network configuration, depicted in Fig. 1, where the UE is connected to the eNB with the help of two relays, is the simplest and most commonly utilized two-relay configuration.

Quantize-Map-Forward (QMF) was recently proposed in information theory as a relaying strategy that allows to achieve a constant gap from the capacity in arbitrary wireless Gaussian networks; the basic idea is that, relays quantize their received signals, perform a random mapping directly to the transmit codebook, and forward the resulting signal to the destination [1]. This work used information-theoretical tools and arguments, such as infinite length coding and exponential decoding complexity[2]; more recently, low complexity schemes were also developed that use LDPC codes to emulate the QMF insights [2], [3]. However, LDPC codes are not part of LTE; moreover, some of these works do not take into account the practical considerations such as integration in the standard LTE operation. In this work we provide, as far as we know, the first method to integrate insights from QMF relaying into LTE-A, using turbo-codes and Hybrid ARQ (H-ARQ).

Our goal is to enable the QMF functionalities at the relays, with minimal changes in the LTE-A standard. Ideally, we would like the existence of relays to be almost transparent to the operation of UE and eNB, *i.e.,* have these experience a "direct" connection to each other that abstracts the intermediary relays. To achieve this, we need to address a number of practical challenges, such as half-duplex operation of nodes and implementation complexity. The challenge with half-duplex radios is on deciding the schedule, namely, when should each relay listen and transmit. We focus on a fixed $1/2$ duty cycle schedule, which effectively creates information flow over two disjoint line networks. We simplify QMF to this network scenario. We implement a scalar quantizer which is matched to the transmit constellation; effectively a demodulator[3]. We call this simplification of QMF as Demodulate-Map-Forward (DMF). We also develop operations that are integrated with the H-ARQ protocol, so that we can directly use standard iterative (H-ARQ) turbo decoders used in LTE systems. In this work we focus on the two-relay diamond network configuration in Fig. 1, however our approach could be extended to the case of the N-relay diamond network.

The paper is organized as follows. Section II presents our system and channel model; Section III discusses the necessary steps towards a system integration for LTE with DMF relaying. Section IV provides our numerical results.

II. System and Channel Models

We consider a half-duplex parallel relay network model where a basestation eNB sends information to a destination UE with the help of two relays, RN1 and RN2, as depicted in Fig. 1. We restrict our attention to downlink (DL) transmissions, in which the information flow is from eNB to UE. We assume there exists no direct link between the source and the destination. We also consider spatially separated relay nodes with no inter-relay interference. All the network nodes are half-duplex, which means that they cannot receive and transmit at the same time.

We are interested in deploying the QMF relaying scheme in the network in Fig. 1. Fig. 2 illustrates the high-level operation of the network components with QMF; the same figure also summarizes our notation. In this diagram, the source employs an encoder followed by a standard LTE rate matcher, which provides a unified method of puncturing/repeating of encoder output. Each relay quantizes its received signal using a quantizer $\mathcal{Q}$, performs a mapping and transmits the resulting signal

[1]This work was partially funded by ERC Project NOWIRE (ERC-2009-StG-240317) and CONECT (FP7-ICT-2009-257616).

[2]This was extended to lattice codes and vector quantization at relays in [4].

[3]This specializes QMF in that the quantization is matched to transmit constellation.

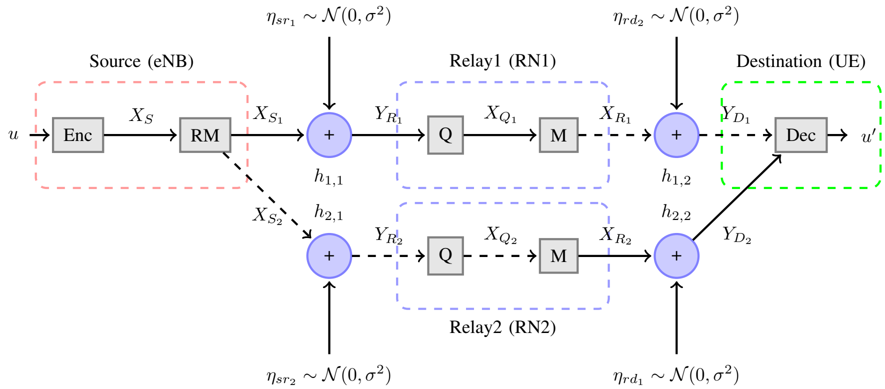


Fig. 2: Modules used to implement DMF in the diamond network configuration of Fig. 1.

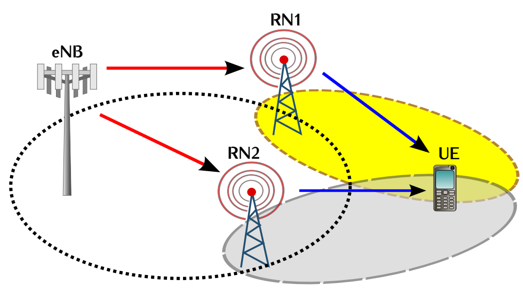


Fig. 1: LTE downlink network operation via relaying on a diamond network: the eNB sends information to two relays, RN1 and RN2, who in turn transmit it to the UE.

to the destination. The destination employs an iterative decoder that uses channel state information.

We assume channels with additive white Gaussian noise (AWGN). The noise variances at each channel (η_{sr_i}, η_{rd_i}) are independently chosen from the same normal distribution ($\mathcal{N}(0,\sigma^2)$). The channel equations for one path (Relay RN*i*) from source to destination in Fig. 2 is given in (1) - (4):

$$Y_{R_i} = h_{i,1} \cdot X_{S_i} + \eta_{sr_i} \quad (1)$$
$$X_{Q_i} = \mathcal{Q}(Y_{R_i}) \quad (2)$$
$$X_{R_i} = M(X_{Q_i}) \quad (3)$$
$$Y_{D_i} = h_{i,2} \cdot X_{R_i} + \eta_{rd_i} \quad (4)$$

where $\mathcal{Q}$ denotes the quantization operation, M a deterministic mapping function. We discuss the specific implementations of these in Section III-D.

III. Steps Towards LTE System Integration

A. Half-Duplex Scheduling

The problem of selecting the optimal schedule for a half-duplex two-relay diamond network, assuming perfect and instantaneous channel knowledge, can be formulated as a linear program, and has been solved in [5]. In our work, we opted for a simple $1/2$ duty cycle scheduling among relay nodes, that has each relay listen and transmit in a complementary fashion. We made this choice for several reasons: in practice, instantaneous channel knowledge is not available; even if it were, dynamic schedules cause additional complexity, since, depending on the channel realizations, the decoding operations would need to change; finally, the simple $1/2$ duty cycle is known to be optimal from a diversity-multiplexing tradeoff viewpoint and was shown [5] to universally achieve approximately 88% of the capacity independently of the channel SNRs.

Our relaying schedule is as follows for slotted operation.

- In the *even time slots*, eNB transmits to RN_1 who listens. RN_2 transmits to the UE.
- In the *odd time slots*, eNB transmits to RN_2 and RN_2 to the UE.

This simple schedule was also proposed in [8] in the context of relaying protocols for Amplify-Forward (AF) and Decode-Forward (DF) relays, and has several attractive features:

- Both eNB and UE experience continuous operation (continuous transmission for eNB and continuous reception of UE), as in the conventional point-to-point LTE scenario.
- The information flow is essentially along two edge-disjoint path, with no broadcasting or interference. Thus the decoder at the destination essentially remains the same as in the conventional point-to-point case, with only difference the calculation of its input channel statistics information (the log-likelihood-ratios or LLR values), as we will discuss in the following.

B. Relay operation

Essentially, our half-duplex scheduling results in operating two disjoint paths; thus we opt for a very simple implementation of the QMF $\mathcal{Q}$ and M modules at the relays:

$\mathcal{Q}$: we use symbol level quantization at the transmitted constellation points (thus a demodulation process).

M : in conjunction with the use of H-ARQ, we focus on using the identity mapping[4] (*i.e.,* we simply forward the demodulated symbols using the same constellation).

[4]Since the schedule creates edge-disjoint paths, the relays see different transmit signals motivating the identity map.

This scheme (called DMF) differs from Decode-Forward (DF) in that relays do not decode the codeword before forwarding, and from Amplify-Forward (AF) in that we demodulate, *i.e.,* quantize to the constellation points. In general, DMF can provide more flexibility from a resource allocation perspective compared to AF relays. For example, AF requires to use the same number of resource elements in both hops; in contrast, with DMF, since we demodulate, we can use mappings that reduce or increase the number of transmissions in the second hop.

C. Use of H-ARQ

Our next design challenge is how to avoid modifications on the UE and eNB link layer operation with our system, while still ensuring a reliable link-layer performance. We start this section by describing the standard LTE Hybrid-ARQ (H-ARQ); we then propose two simple algorithms to employ H-ARQ over our network configuration. Each of these algorithms achieves a different throughput-delay tradeoff, that we evaluate in Section IV.

Standard LTE H-ARQ: At the link layer, for each given packet that eNB wishes to send to UE, the standard LTE H-ARQ produces a codeword and then uses a rate matching (RM) process to generate (up to) 4 redundancy versions (RV) of this codeword [7]. A common method to produce RVs in LTE-HARQ is *Incremental Redundancy* (IR), in which each RV contains a different set of punctured/repeated bits from the same codeword (s_i). Each RV is self-decodable at the destination, and at the same time jointly decodable with other RV of the same codeword. The destination attempts to decode a received RV (say $s_i(0)$); if this is not possible, this is stored, and when the destination receives $s_i(1)$ of the same codeword, it is combined with $s_i(0)$ and joint decoding is attempted; similarly if more RVs are needed. A transmission of a different RV is called a *retransmission* while the combining at the decoder is called *soft-combining*.

In the H-ARQ *stop-and-wait mode*, the transmitter waits for a receiver acknowledgment (ACK) before continuing with the retransmissions. Stop-and-wait is attractive since only necessary RVs are retransmitted. To avoid loss in efficiency while waiting for the ACK, multiple stop-and-wait H-ARQ processes are activated in parallel: while one H-ARQ process is waiting for an ACK, another process sends information to the channel.

Integration of LTE H-ARQ to parallel relay networks: We propose two straightforward adaptations of LTE-HARQ (algorithm 1 and 2), that decide the scheduling of RVs, i.e., which RV the eNB transmits towards which relay, and when. These algorithms require minimal changes to the link layer operation, thanks to the following fact: in LTE Downlink (DL), H-ARQ is *asynchronous*, and in particular the schedule for RV retransmissions is not predefined. As a result, the transmitted RV blocks already contain control information such as the H-ARQ process ID, a new transmission or retransmission flag. This inherent flexibility leaves the door open for scheduling while relaying. Essentially, at the link layer of the eNB, what would differentiate relaying from a point-to-point connection would be the larger delay to collect ACKs, and thus the need to increase the number of processes that run in parallel.

We use the following notation to describe the two algorithms; we also provide illustrating examples in Table Ia and Ib.

- $rv \in \{0, 1, 2, 3\}$.
- $s_i(rv)$: Redundancy version (rv) of message s_i.
- $\tilde{s}_i(rv)$: Received (noisy) version of $s_i(rv)$ at a relay.
- $\breve{s}_i(rv)$: Received (noisy) version of $s_i(rv)$ at the UE.

Algorithm 1 (Proactive): In this scheme the source (eNB), when a new packet transmission (s_i) is scheduled, sends consecutively two redundancy versions ($s_i(0)$ & $s_i(1)$), that reach the destination through RN1 and RN2. If decoding with both these RVs is not possible, the source will send next $s_i(2)$ and $s_i(3)$. That is, the source always sends a batch of 2 *rv*s of a packet, in the interest of minimizing delay, even though the packet s_i might be decodable with one RV.

Algorithm 2 (Stop-and-wait): In this scheme, the eNB sends only one RV per packet (e.g. $s_0(0)$ for packet s_0), and waits to collect an ACK/NACK before retransmissions (of $s_0(1)$, $s_0(2)$, $s_0(3)$). While waiting, additional active H-ARQ processes attempt to send RV for other packets (e.g. s_1, s_2, etc.).

D. Encoding and Decoding

LTE specifies the use of turbo encoders and decoders for the downlink channels. A turbo decoder takes as input log-likelihood-ratios (LLR) that are calculated from the channel observations assuming point-to-point links. To be compatible, we propose to use the specified encoders and decoders, but recompute the LLR of the received bits at the decoder input taking into account the existence and operation of the relays.

For the rest of the section we assume binary signaling (e.g. BPSK) at the source and relays, although these computations are also valid for QPSK. Computations for QAM constellations can be done in a similar fashion. We assume that both relays are DMF relays that operate as we described in Section III-B and Fig. 2, namely, quantize the received signal to the closest point of the source signal constellation.

Consider a transmitted bit X_{S_i}, and the associated LLR:

$$LLR_{X_{S_i}} = \ln\left(\frac{P(Y_{Q_i} \mid X_{S_i} = +1)}{P(Y_{Q_i} \mid X_{S_i} = -1)}\right) \quad (5)$$

In [9] we show that (5) can be approximated as:

$$LLR_{X_{S_i}} \approx \text{sign}(Y_{D_i} h_{i;2}) \ \min(\frac{|2Y_{D_i} h_{i;2}|}{\sigma^2}, \frac{|h_{i,1}|^2}{4\sigma^2}) \quad (6)$$

where the sign() function returns +1 or -1 depending on the polarity of the value of its parameter, and the notation is given in Fig. 2 and (1)-(4). The precise bounds for the approximation are given in [9]. The decoder can implement (6) through the following simple steps: Compare the compensated signal ($|2Y_{D_i} h_{i;2}|$) and the magnitude of the first channel ($|h_{i,1}|^2/4$). Use the minimum of them as the magnitude of $LLR_{X_{S_i}}$, and use the sign($|2Y_{D_i} h_{i;2}|$) as the sign of $LLR_{X_{S_i}}$.

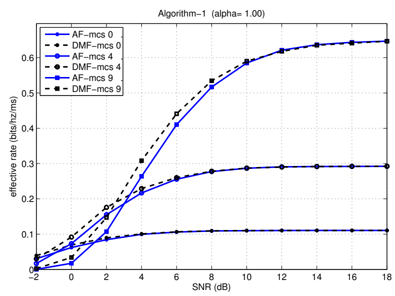
TABLE I: Two H-ARQ algorithms for the parallel relay network. Columns represent the timeslots for DL communication (we assume only downlink) and rows represents the receive (rx) or transmit (tx) behavior of each network element.

(a) Algorithm 1 (Proactive)

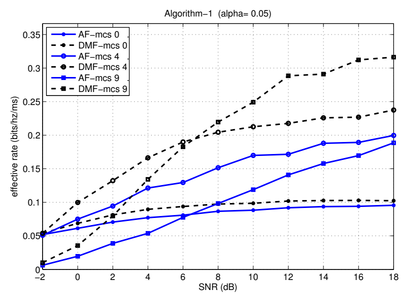
	Time $\longrightarrow$						
eNB_{tx}	$s_1(0)$	$s_1(1)$	$s_2(0)$	$s_2(1)$	$s_1(2)$	$s_1(3)$	$s_1(4)$
$R1_{rx}$	$\tilde{s}_1(0)$	·	$\tilde{s}_2(0)$	·	$\tilde{s}_1(2)$	·	$\tilde{s}_1(4)$
$R1_{tx}$	·	$\tilde{s}_1(0)$	·	$\tilde{s}_2(0)$	·	$\tilde{s}_1(2)$	
$R2_{rx}$	·	$\tilde{s}_1(1)$	·	$\tilde{s}_2(1)$	·	$\tilde{s}_1(3)$	·
$R2_{tx}$	·	·	$\tilde{s}_1(1)$	·	$\tilde{s}_2(1)$	·	$\tilde{s}_1(3)$
UE_{rx}	·	$\breve{s}_1(0)$	$\breve{s}_1(1)$	$\breve{s}_2(0)$	$\breve{s}_2(1)$	$\breve{s}_1(2)$	$\breve{s}_1(3)$
Decode?	·	×	×	×	✓	✓	?

(b) Algorithm 2 (Stop-and-wait)

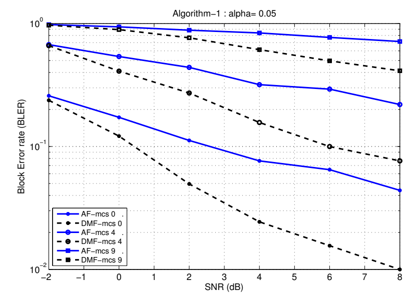
	Time $\longrightarrow$						
eNB_{tx}	$s_1(0)$	$s_2(0)$	$s_1(1)$	$s_2(1)$	$s_1(2)$	$s_3(0)$	$s_4(0)$
$R1_{rx}$	$\tilde{s}_1(0)$	·	$\tilde{s}_1(1)$	·	$\tilde{s}_1(2)$	·	$\tilde{s}_4(0)$
$R1_{tx}$	·	$\tilde{s}_1(0)$	·	$\tilde{s}_1(1)$	·	$\tilde{s}_1(2)$	·
$R2_{rx}$	·	$\tilde{s}_2(0)$	·	$\tilde{s}_2(1)$	·	$\tilde{s}_3(0)$	·
$R2_{tx}$	·	·	$\tilde{s}_2(0)$	·	$\tilde{s}_2(1)$	·	$\tilde{s}_3(0)$
UE_{rx}	·	$\breve{s}_1(0)$	$\breve{s}_2(0)$	$\breve{s}_1(1)$	$\breve{s}_2(1)$	$\breve{s}_1(2)$	$\breve{s}_3(0)$
Decode?	·	×	×	×	✓	✓	✓



(a) Rate : Symmetric channel network

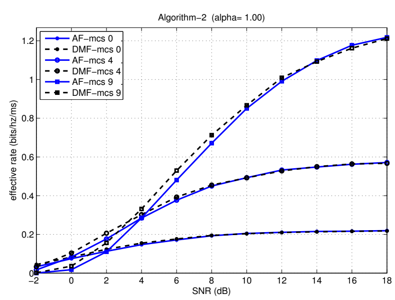


(b) Rate: Asymmetric channel network

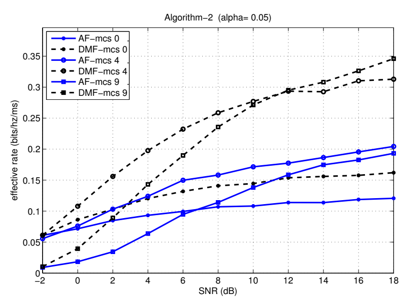


(c) BLER: Asymmetric channel network

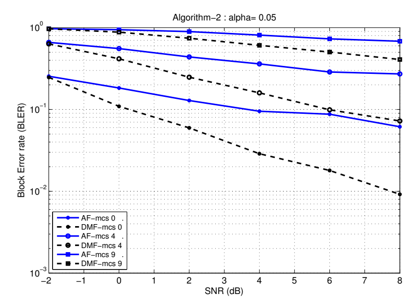
Fig. 3: Algorithm 1



(a) Rate : Symmetric channel network



(b) Rate: Asymmetric channel network



(c) BLER: Asymmetric channel network

Fig. 4: Algorithm 2

IV. Simulation Environment & Results

We compare DMF relaying (as described in Section III-B) with an AF scheme, that has the relays, instead of demodulating, simply forward their received signals.

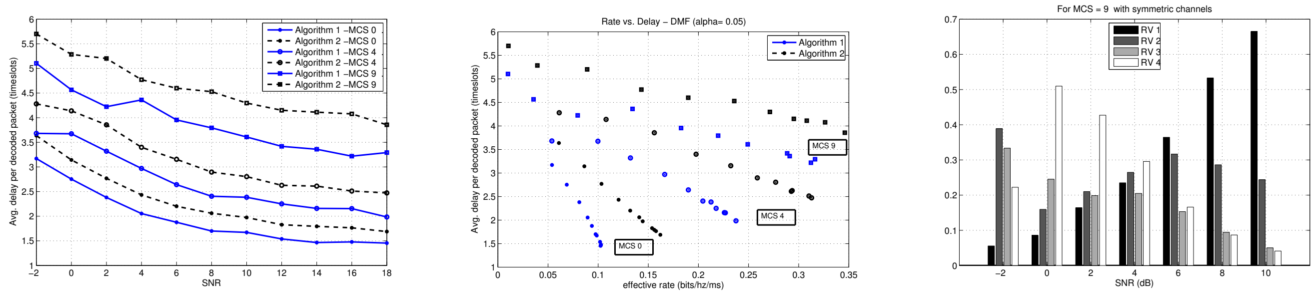
A. Simulation Environment

We deployed DMF and AF relaying on EURECOM's long term evolution (LTE) compliant OpenAir Interface(OAI) platform [6]. OAI platform uses the standard LTE transmitter (and receiver) blocks as defined in the 3GPP LTE Rel. 8 specifications [7]. In our experiments, we used link-level simulators provided by OAI for downlink channel, and added some relaying functionality to the existing simulator.

For the channel simulations, we used standard OAI single-tap Rayleigh channel models. We assumed fully loaded H-ARQ processes; that is, we assumed there is always enough number of bits provided from upper layers for each H-ARQ process. We also assume error-free ACK/NACK reception.

Source & Destination: We used standard LTE eNB and UE nodes for LTE downlink transmission. We provide in the following some details.

- For channel coding at the DLSCH(DownLink Shared CHannel) the standard 1/3-rate Turbo encoder is used [7].
- The effective code rate is matched for transmission depending on the received MCS (Modulation-Coding-Scheme) value (Table II). MCS is chosen according to the CQI (Channel Quality Indicator) received from the destination. In other words, the estimated DL (downlink) channel quality decides the modulation and the effective



(a) Delay: Asymmetric channel network

(b) Rate vs. Delay: Asymmetric channel network

(c) Distribution of H-ARQ rounds at which the packet is decoded: Symmetric channel network

Fig. 5: Various performance comparisons of two algorithms with DMF relaying

TABLE II: MCS values used in simulations.

MCS Index	Modulation	Effective. Cod. Rate
0	QPSK	0.1079365
4	QPSK	0.2857143
9	QPSK	0.6361905

coding rate. The output of the turbo encoder is then rate-matched to the effective rate. In our simulations MCS is a parameter to the simulation and does not change for the life of a single simulation.

- The turbo decoder at the destination calculates the bit LLR values as it does for standard LTE decoder when we use AF relays. However, bit LLRs are recomputed for DMF relays as discussed in Section III-D.

Channel network configurations : A *symmetric channel network* refers to a diamond relay network in which all the 4 channels have symmetric channel statistics (same SNR). An α*-asymmetric channel network* refers to a diamond network in which the channels eNB-RN1 and RN2-UE have the same SNR, while the remaining channels eNB-RN2 and RN1-UE have an α fraction of the SNR (i.e,. $\alpha \cdot SNR$). In our simulations, we chose $\alpha = 0.05$. We assume static channels and that the channel coefficients are known at the destination.

Metrics: We will use the following performance metrics: *Effective rate* refers to the rate of successfully transmitted bits and is measured in bits/sec/Hz. *Block-error-rate (BLER)* is defined as the ratio of unsuccessful blocks all four RVs are transmitted, divided by the total number of transmitted blocks. *Delay* refers to number of transmitted LTE blocks; Algorithm 1 transmits two RVs within one such block, while Algorithm 2 would send these in consecutive blocks (after receiving feedback). MCS values correspond to different coding rates as used in the LTE standard, and are summarized in Table II.

B. Results

Fig. 3a and 3b depict the effective rates *Algorithm 1* achieves for symmetric and asymmetric channel conditions, respectively; similarly, Fig. 4a and 4b depict the performance of *Algorithm 2*. In both cases, we observe that DMF relaying performs slightly better than AF for the symmetric case and provides significant performance gains for asymmetric networks. Fig. 3c and Fig. 4c depicts the BLER performance of Algorithm 1 and 2, respectively, for the asymmetric networks. We again see the advantage that DMF offers for a range of coding rates (mcs values). Moreover, Algorithm 2 offers a significant throughput improvement as compared to Algorithm 1, especially at the high SNR regimes. This implies that the additional RV that Algorithm 1 transmits in a significant number of cases are not needed for decoding. Indeed, the histogram in Fig. 5c plots the fraction of cases that 1, 2, 3 or 4 RV are needed for decoding: we can see that as SNR increases, in most cases a single RV is sufficient for decoding.

Fig. 5a compares both algorithms in terms of their decoding delay. As expected, Algorithm 1 leads to lower delay as compared to Algorithm 2, thanks to being proactive in terms of sending RVs. Fig. 5b depicts the throughput vs. delay tradeoff of both algorithms for different coding rates.

ACKNOWLEDGMENT

The authors also would like to acknowledge the stimulating discussions with Dr. E. Yılmaz and A. Sengupta.

REFERENCES

[1] S. Avestimehr, S. Diggavi and D. Tse, “Wireless Network Information Flow: A deterministic approach”, ArXiv e-prints, arXiv:0906.5394v7

[2] V. Nagpal, I-Hsiang Wang, M. Jorgovanovic, D. Tse, B. Nikolic, “Quantize-Map-and-Forward Relaying: Coding and System Design”, Proc. of the Annual Allerton Conference on Comm., September 2010.

[3] A. Sengupta, S. Brahma, A. Ozgur, C. Fragouli, S. Diggavi, “Graph-based Codes for Quantize-Map-and-Forward Relaying.”, IEEE Information Theory Workshop (ITW), 2011.

[4] A. Ozgur and S N. Diggavi. Approximately achieving Gaussian relay network capacity with lattice codes. *Proc. of IEEE ISIT 2010, Austin, Texas*, pp 669–673, June 2010. See also http://arxiv.org/abs/1005.1284.

[5] S. Brahma, A. Ozgur, C. Fragouli , “Simple Relaying Strategies for Half-Duplex Networks.”, IEEE International Symposium on Information Theory (ISIT), 2012.

[6] http://www.openairinterface.org

[7] *3GPP LTE*, Standard 36.212, Rev. 8.5.0, Feb. 2009.

[8] B. Rankov, and A. Wittneben,“Spectral efficient protocols for half-duplex fading relay channels.”, IEEE Journal on Selected Areas in Communications, vol. 25, 379–389, 2007.

[9] E. Atsan et al. EPFL Technical Report, EPFL-REPORT-180103, 2012.