

Recent advances in HARQ communications

— Tutorial to be presented at ICT 2019, Hanoi —

Presented by Pierre Duhamel (CNRS/CentraleSupélec/L2S, France)
Co-authors : Leszek Szczecinski (INRS, Canada), Philippe Ciblat
(Telecom ParisTech, France) and Francesca Bassi
(CNRS/CentraleSupélec, France)

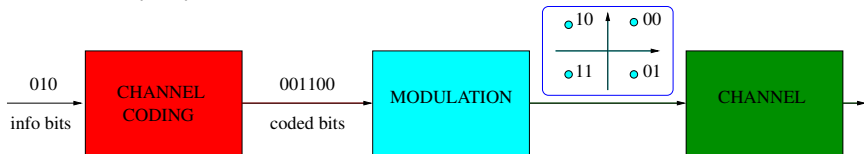
April, 2019

**With many
contributions from**

Faton Maliqi, Alaa Khreis, Mohamed Jabi

Context : (Short) description of a simplified wireless commutation scenario

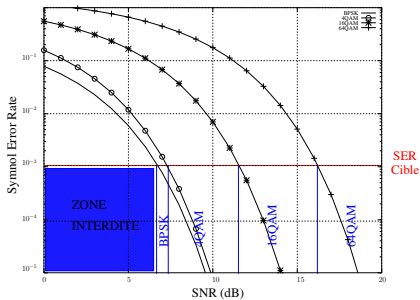
Transmitter (TX):



Traditional presentation :

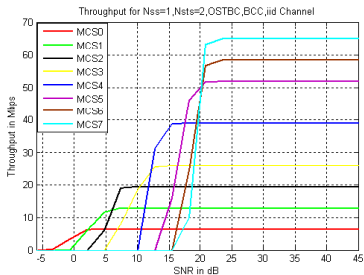
- Adaptive Modulation and Coding: adapts the amount of information transmitted to the "quality" of the channel
 - obviously requires the transmitter to know the channel parameters
 - and to have a performance model for the considered channel
- Transmitter does not know if the transmission failed

Example: AMC with QAM modulation



In actual situations : there exists a target error rate...

Another example: AMC with QAM modulation in 802.11n



MCS in 802.11n, by Meifang Zhu, MSc @ EIT

Drawbacks

- Not many degrees of freedom in the design of AMC
- Would require full knowledge of the instantaneous channel parameters
- When used with average channel conditions, lack of adaptivity (true propagation conditions, noise level)

Note also that practical implementations require anyway a feedback channel :

The receivers estimates the "quality" of the channel (usually the SNR) , and sends it back to the transmitter, which is then transmitting with the most appropriate Modulation and Coding Scheme (MCS)

Part 1 : The general picture

However, this is a pure "Physical Layer" point of view, and there could be many problems in the interactions between the various ingredients of a wireless communication network...

Therefore, we spend some time in giving an overview of the aspects that are strongly interconnected... (in order to propose the smartest HARQ...)

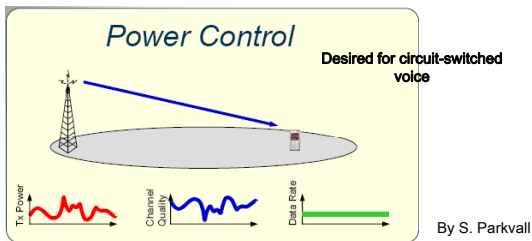
Motivation

- Rapid varying radio channel
 - Time-variant: coherence time (Doppler spread)
 - Frequency-selective: coherence bandwidth (delay spread)
 - Interference
- Exploit the channel variation *prior to* transmission
 - Link adaptation : Set transmission parameters to handle radio channel variation
 - Channel-dependent scheduling: Efficient resource sharing among users
- Handle the channel variation *after* transmission
 - Hybrid ARQ : Retransmission request of erroneously received data packets

Link adaptation (1)

Power control:

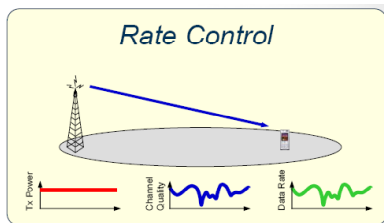
- Dynamically adjust the transmit power to compensate for the varying radio channel condition
- Maintain a certain SNR at the receiver
- Constant data rate regardless of the channel variation



Link adaptation (2)

Rate control:

- Packet-data traffic: not a strong desire for constant rate (as high rate as possible)
- Dynamically adjust the data rate to compensate for the varying radio channel condition
- Full constant transmit power (desirable in multiuser systems)



By S. Parkvall

Link adaptation (3)

- Rate control
 - Adaptive Modulation and Coding(AMC) scheme
 - "Good" channel condition: Bandwidth limited (High-order modulation + high-rate coding)
 - "Poor" channel condition: Power limited (Low-order modulation + low-rate coding)
- In HSDPA link adaptation
 - QPSK for noisy channels and 16 QAM for clearer channels
 - 14Mbps, on clear channels using 16-QAM and close to 1 coding rate.
 - 2.4 Mbps, on noisy channels using QPSK and 1/3 coding rate (14 Mbps \times 1/2 \times 1/3)
 - This adaptation is performed up to 500 times per second

Link adaptation (4)

- Power control: constant rate
 - Desired for voice/video (Short-term rate variation not an issue with constant average data rate)
 - Inefficient use of transmit power
- Rate control: constant (max) transmit power
 - Adaptive data rate
 - Efficient use of transmit power
 - Desired in multiuser systems to reduce variations in interference power

[Chung & Goldsmith, 2001] Little spectral efficiency is lost when the power or rate is constrained to be constant, with optimal adaptation.

Scheduling

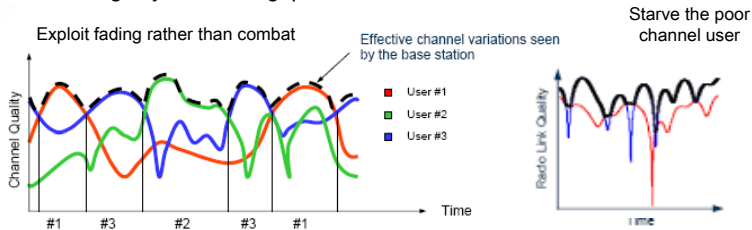
- The allocation of the shared resources among the users at each time instant
 - Whom?
 - How?
- Joint function with link adaptation
- Channel dependent
- Downlink scheduling \Rightarrow Centralized resource
- Uplink scheduling \Rightarrow Distributed resource

Two examples below of extreme choices for Downlink scheduling, and a more reasonable one (we do not consider uplink in this context description...)

Downlink Scheduling (1)

- Channel-dependent scheduling
 - Max-C/I (Max rate) scheduler
 - Schedule at the fading peaks
 - Independently varying radio links
 - Multiuser diversity gain
 - High system throughput but not fair

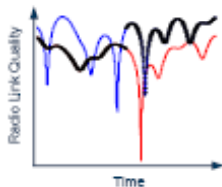
$$k = \arg \max_i R_i$$



By S. Parkvall

Downlink Scheduling (2)

- Round-robin scheduling
 - Regardless of channel conditions
 - Fair? ... same amount of the radio resources
 - Unfair! ... service quality (more resources needed for poor channel)
 - Simple but poor performance



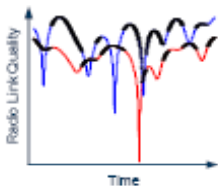
By S. Parkvall

Downlink Scheduling (3)

- Two-fold requirement
 - Take advantage of the fast channel variations
 - Ensure the same average user throughput
- Proportional-fair scheduler
 - Proportion between the instantaneous data rate and the average data rate during a certain period
 - High throughput and fairness

$$k = \arg \max_i \frac{R_i}{\bar{R}_i}$$

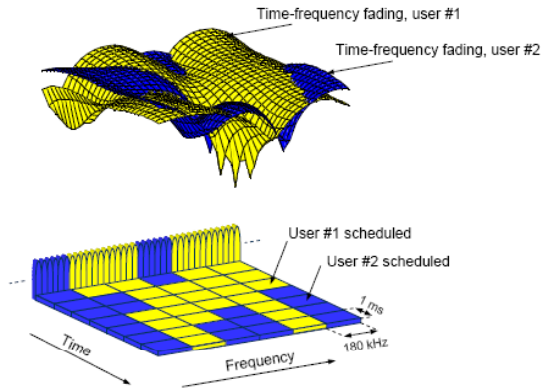
Schedule on fading peaks,
regardless of the absolute quality



By S. Parkvall

Downlink Scheduling (4)

- LTE
 - channel-dependent scheduling in time and frequency domains



By S. Parkvall



Requirements on Channel state information

In what follows, we implicitly work with Block fading channels : even if an average situation is safe, very bad channels may occur...

- CSI : Needed at TX for link adaptation and channel-dependent scheduling
- Downlink
 - Pilot signal ? e.g., Correlation channel estimator
 - Measured channel conditions reported to BS => Outdated if high mobility
 - Channel prediction : Additional complexity and constraint
 - Link adaptation based on " long-term" average channel

How to adapt to channel's variation? : from AMC to ARQ

Summary : advanced packet radio wireless networks such as HSDPA, channel-dependent scheduling may be used to take advantage of favourable channel conditions to increase the throughput and system spectral efficiency ... (wireless communications are a very "liberal" situation: efficient channels / users should be used as much as possible)

- Since AMC is working with average (non instantaneous) performance,
- Idea: trial and error
 - First send a packet of symbols
 - if correctly received (ACK), 
 - if residual errors (NACK),  and send again a packet containing "same" information...
- This requires feedback channel : information on the instantaneous channel, and the success of the transmission.

.... and do not forget that there is delay in the feedback : processing time, transmission time, framing time, etc...

Part 2 : Classical ARQ/HARQ situations

- ARQ
- HARQ
- HARQ taxonomy :
 - Type I and II
 - Chase Combining, Incremental Redundancy

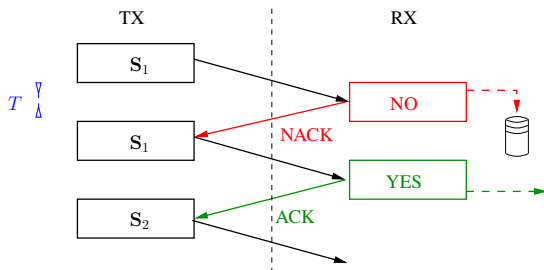
And we first assume that everything is instantaneous

ARQ (*Automatic ReQuest*) overview : the ingredients

- Forward Error Correction (FEC)
 - Add redundancy for error correction
- Automatic Repeat Request (ARQ)
 - Compatible with TCP behavior for packet data
 - Error-detecting code by Cyclic Redundancy Check (CRC)
 - CRC used as a check sum to detect errors (Division of polynomials in Galois field $GF(2)$...remainder...)
 - No error? Positive acknowledgement (ACK)
 - Error? Negative acknowledgement (NAK)
- Hybrid ARQ
 - Combination of FEC and ARQ
 - FEC: correct a subset of errors
 - ARQ: if still error detected

From ARQ (*Automatic ReQuest*) ...

Let $\mathbf{S} = [s_0, \dots, s_{N-1}]$ be a packet composed by N uncoded symbols



Assume for a while that all processing

- transmission from TX to RX
- Processing at TX
- Travel time for feedback from RX
- Additional processing at TX

is instantaneous....

... Towards Hybrid ARQ (HARQ): Type-I HARQ

Remark

Retransmission does not contradict forward error coding (FEC)

Type-I HARQ: packet \mathbf{S} is composed by coded symbols s_n

- first packet is more protected
- there is less retransmission
- transmission delay is reduced

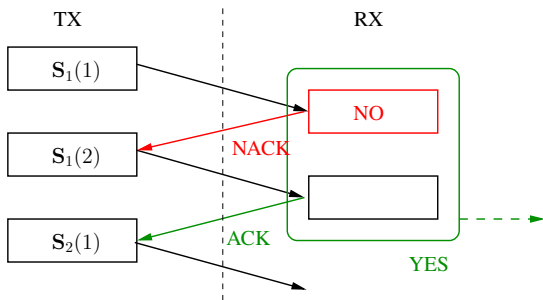
- Efficiency is upper-bounded by the code rate

Drawbacks

- Each received packet is treated independently
- Mis-decoded packet is thrown in the trash

Type-II HARQ

Memory at RX side is considered \Rightarrow Type-II HARQ



Main examples:

- *Chase Combining* (CC)
- *Incremental Redundancy* (IR)

Examples: CC-HARQ and IR-HARQ

CC

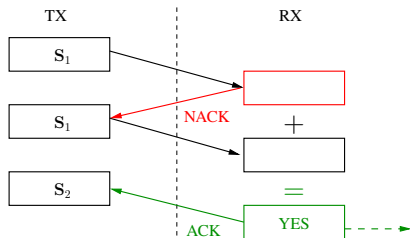
$$Y_1 = S_1 + N_1$$

$$Y_2 = S_1 + N_2$$

then detection on

$$Y = (Y_1 + Y_2)/2$$

SNR-Gain equal to 3dB



IR

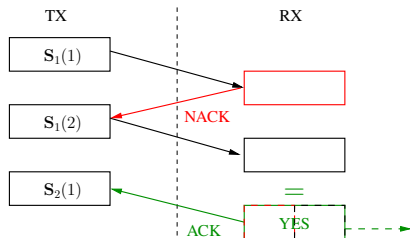
$$Y_1 = S_1(1) + N_1$$

$$Y_2 = S_1(2) + N_2$$

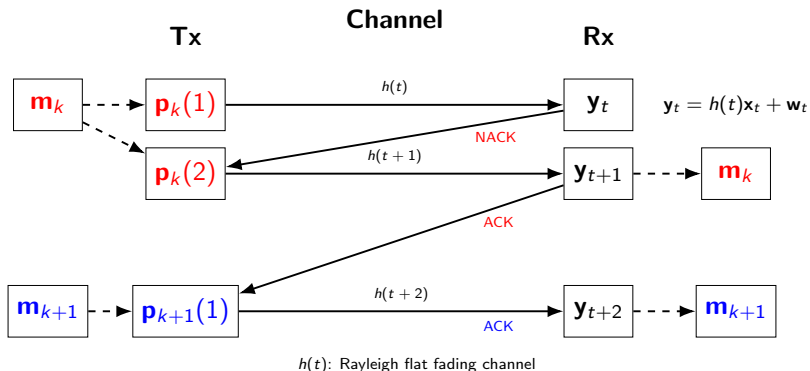
then detection on

$$Y = [Y_1, Y_2]$$

Coding gain



Hybrid ARQ (Automatic Repeat reQuest)



$\mathbf{p}_k(\ell)$: ℓ -th packet of message \mathbf{m}_k , $\ell \in \{1, \dots, C\}$

$\mathbf{p}_k(1) = \mathbf{p}_k(2)$

for CC-HARQ (Chase Combining)

→ diversity gain

$\mathbf{p}_k(1) \neq \mathbf{p}_k(2)$

for IR-HARQ (Incremental Redundancy)

→ diversity + coding gain

Part 3 : Performance metrics

- **Packet Error Rate (PER):**

$$\text{PER} = \text{Prob}(\text{information packet is not decoded})$$

- **Efficiency** (*Throughput/Goodput/etc*):

$$\eta = \frac{\text{information bits received without error}}{\text{transmitted bits}}$$

- **(Mean) delay:**

$$d = \# \text{ transmitted packets when information packet is received}$$

- **Jitter:**

$$\sigma_d = \text{delay standard deviation}$$

Quality of Service (QoS)

- Data: PER and efficiency
- Voice on IP: delay
- Video Streaming: efficiency and jitter

Part 4 : Degrees of freedom in the design of HARQ

HARQ in its context : which tools would allow for some improvement ?

4.1 Power adaptation

4.2 Bandwidth adaptation

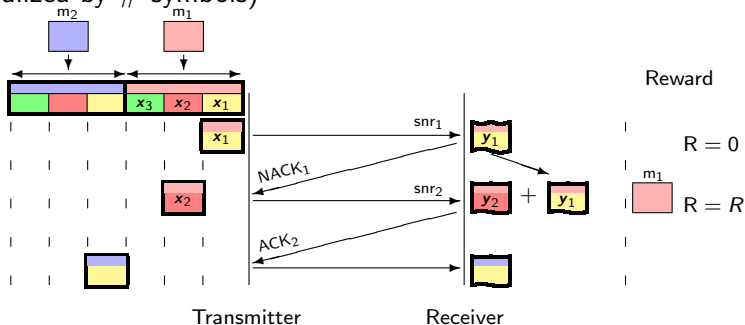
4.3 Rate (reward) adaptation

4.4 Layered coded HARQ

4.5 Non orthogonal HARQ; reducing the delay and improving the throughput (Pierre)

Back to Basics: Canonical HARQ (fixed rate, Rayleigh)

Figure below explicits the subcodewords, and the reward ($\#$ bits, normalized by $\#$ symbols)

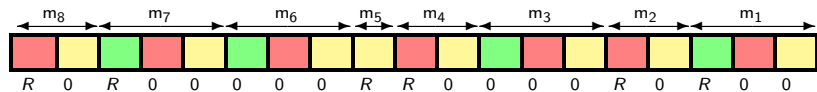


“Canonical” model, to be questioned below

- Constant power
- Constant bandwidth
- Binary reward $R \in \{0, R\}$

Renewal-Reward Theorem

variable bandwidth (multiple rounds) + variable reward (final NACK)



Throughput

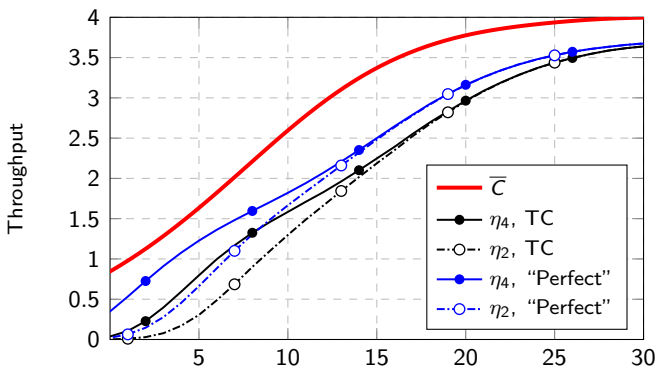
$$\begin{aligned} \eta_K &\triangleq \lim_{T \rightarrow \infty} \frac{1}{T} \sum_{t=1}^T R(t) = \frac{\mathbb{E}[R]}{\mathbb{E}[D]} \\ &= \frac{R(1 - f_1) + R(f_1 - f_2) + \dots + R(f_{K-1} - f_K)}{(1 - f_1) + 2(f_1 - f_2) + \dots + (K - 1)(f_{K-2} - f_{K-1}) + Kf_{K-1}} = \frac{R(1 - f_K)}{1 + \sum_{k=1}^{K-1} f_k}, \end{aligned} \quad (1)$$

Sequence of / decoding errors

$$f_l \triangleq \mathbb{P}\{\text{NACK}_l\}$$

Example: 16-QAM, Rayleigh fading, $R = 3.75$, $K \in \{2, 4\}$

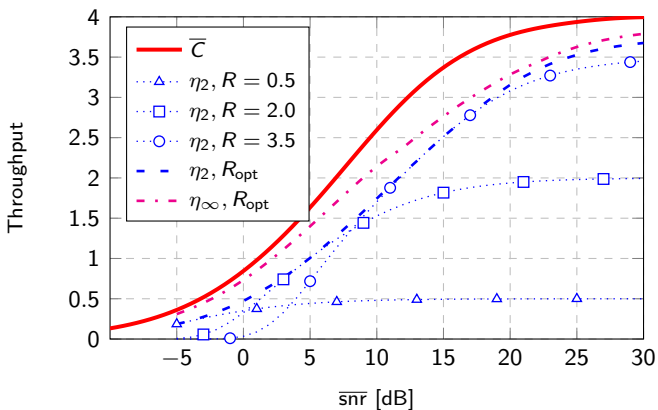
Turbo-codes, fixed rate, varying $T = 2, 4$



Incremental redundancy

- Shannon bounds predict well the performance of practical codes; throughput grows with K
- Gains appear in "low" throughput, i.e., for $\eta_K < R$
- No/negligible gains for "high" throughput $\eta_K \approx R$ (obvious !)

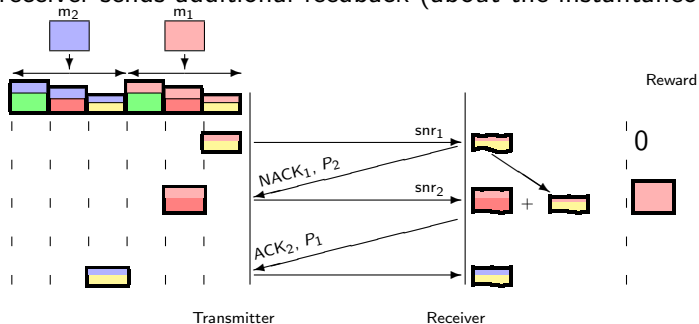
Example: adjusting the rate $R \in \{0.25, 0.5, \dots, 7.75\}$



- Throughput can be improved adjusting R , but
- No significant gains even for $\eta_K \approx R$ even when for $K = \infty$
- Theoretical result: $\lim_{K \rightarrow \infty} \eta_K = \bar{C}$, but only if $R \rightarrow \infty$ (not practical!)

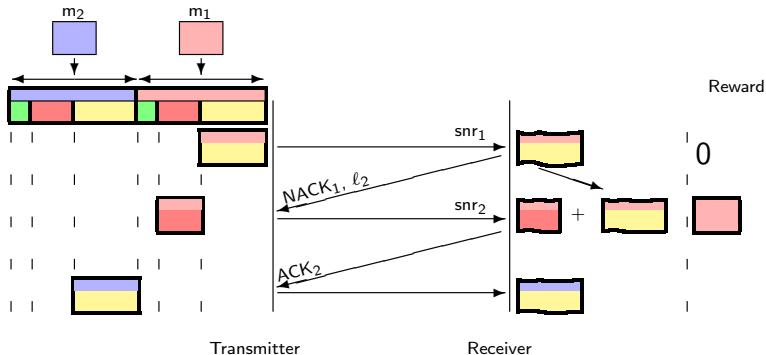
Power adaptation

The receiver sends additional feedback (about the instantaneous SINR)



- The transmitter varies the power of the subcodeword in each round
- The sub-codewords have the same length
- Adaptation: power varies according to the *extra* feedback (precalculated function)
- Allocation: power varies according to the index of the round (precalculated scalar)

Length adaptation



- The transmitter varies the bandwidth (e.g., length) $N_{s,k}$ in each round
- $l_k \triangleq N_{s,k}/N_{s,1}$ is the the normalized bandwidth; $l_1 = 1$.
- Transmission with constant power, $P_k = 1 \forall k$
- Adaptation: bandwidth varies according to the *extra* feedback (precalculated function)
- Allocation: bandwidth varies according to the index of the round (precalculated scalar)

Throughput

Variable power HARQ

$$\eta_K^{\text{VP}} = \frac{R(1 - f_K)}{1 + \sum_{k=1}^{K-1} f_k} \quad (2)$$

constraint (in allocation):

$$\bar{P} = \frac{\sum_{k=1}^K P_k (f_{k-1} - f_k)}{1 + \sum_{k=1}^{K-1} f_k} \quad (3)$$

Variable length HARQ

$$\eta_K^{\text{VL}} = \frac{R(1 - f_K)}{1 + \bar{\ell}}, \quad (4)$$

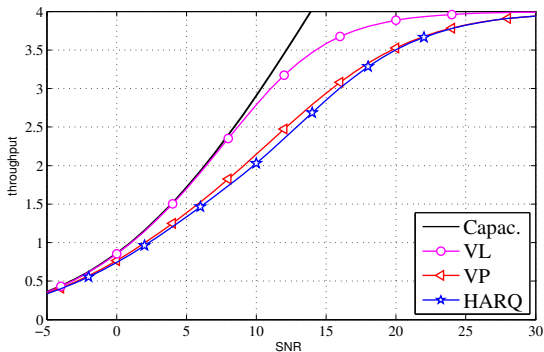
where (for allocation)

$$\bar{\ell} = \sum_{k=2}^K \ell_k f_{k-1}$$

In both cases, the reward (rate) does not change, only the expression of the constraint...

VL vs. VP example: “Shannon codes”, Rayleigh, $R = 4$

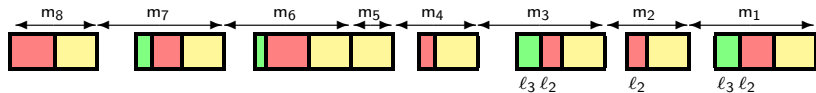
After optimization (using dynamic programming)



- Adaptive power does not help throughput (but can decrease packet loss)
- Adaptive bandwidth yields significant gains in terms of throughput

Variable bandwidth HARQ

Now let us do the converse w.r.t. eq. 4: fix bandwidth, make full use of it, and check what happens on the reward



System level considerations

- Manage “empty” space within the block via
 - frequency allocation (4G) or
 - use of many packets within a single block
- Potential issues: increased signaling overhead and optimization problem.

Reward/rate adaptation

Manipulating the term in the numerator of the throughput expression

- Fixed reward

$$\eta_K = \frac{R(1 - f_1) + R(f_1 - f_2) + \dots + R(f_{K-1} - f_K)}{1 + \sum_{k=1}^{K-1} f_k}$$

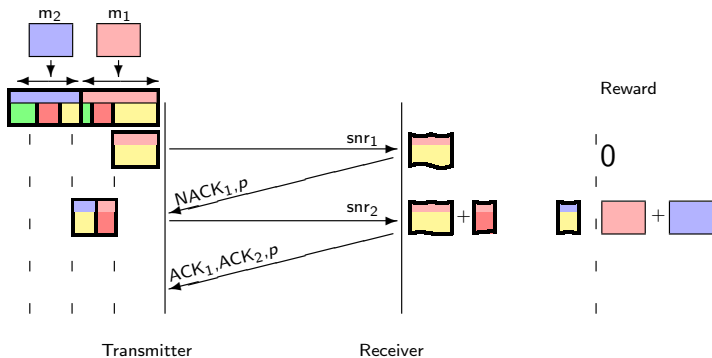
- Variable reward

$$\eta_K = \frac{R(1 - f_1) + R_2^{\Sigma}(f_1 - f_2) + \dots + R_K^{\Sigma}(f_{K-1} - f_K)}{1 + \sum_{k=1}^{K-1} f_k}$$

Notation : R_2^{Σ} : accumulated reward with 2 transmissions

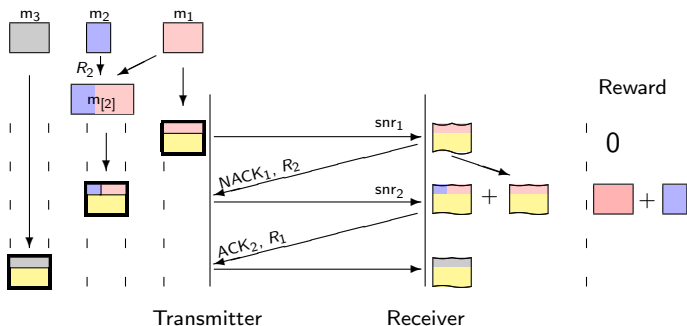
- Interpretation: multi-packet transmission per round
 - Proposition 1: Time-sharing (TS)
 - Proposition 2: Cross-packet coding (XP)

Time Sharing HARQ



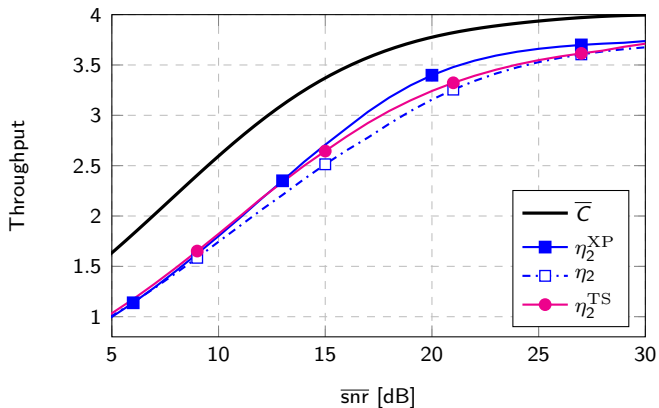
- Time sharing: the sub-codewords of two packets are transmitted in non-overlapping manner.
- p the portion of time/bandwidth allocated to different packets in the same block (depends on the outdated snr)

Cross-packet coding HARQ



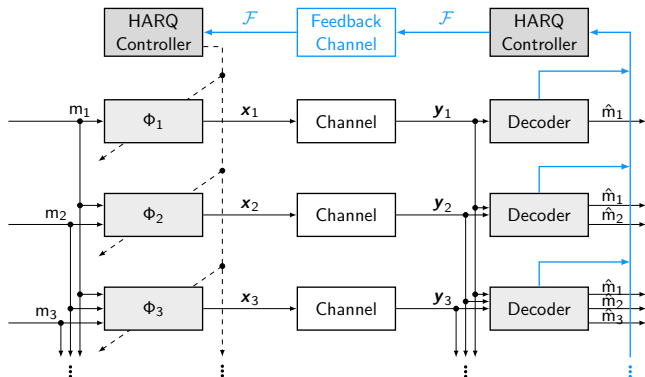
- R_1 is used in the first round, i.e., $m_1 \in \{0, 1\}^{R_1 N_s}$.
- $m_{[2]} = [m_1, m_2] \in \{0, 1\}^{(R_1 + R_2) N_s}$ is encoded using a conventional code.

Example: XP vs TS; 16QAM, Rayleigh fading



Cross-packet (XP) coding is the winner but...

XP-HARQ: encoding, decoding and reward/rate adaptation



Φ_i : encoder for packet i ;

m_{i+1} is jointly encoded with $m_i \implies$ encoder and decoder become increasingly complex

Cross-packet coding: practical issues

Issues with encoding

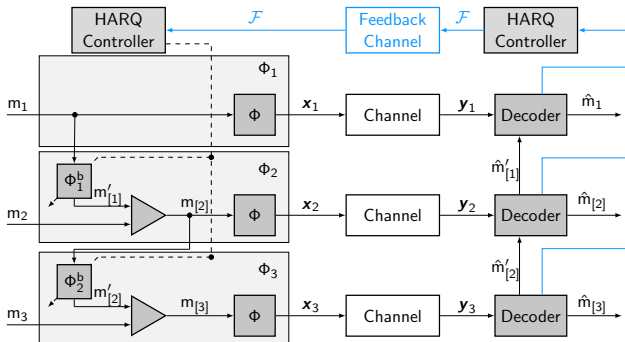
- Growing size of inputs $m_{[k]} = [m_1, \dots, m_k] \in \{0, 1\}^{N_s R_k^\Sigma}$
- Sub-optimality of the encoder design, due to the growing rate, e.g., R_k exceed constellation size, concatenation of codes, etc.

Issues with decoding

- Joint decoding (on multiply-concatenated codes); possible but non-standard
- Multidimensional-multiparametric PER curves (surfaces) are hard to measure, store, and use (for adaptation)

$$\mathbb{P}\{\text{NACK}_k\} = \text{PER}(\text{snr}_1, \dots, \text{snr}_k; R_1, \dots, R_k)$$

Layer-coded HARQ (L-HARQ)

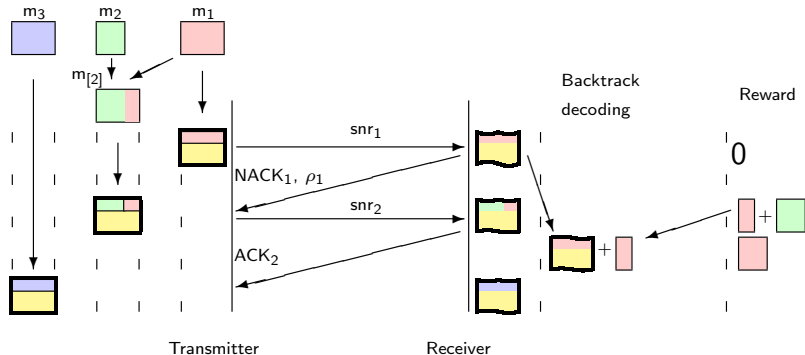


Practical implementation of XP-HARQ

- now : same encoder Φ (same # of bits at the inputs)
- Multipacket encoding \rightarrow puncturing (with rate ρ) and binary packet mixing + Off-the-shelf (optimized) encoder
- Practically : transmit part of m_1 (punctured) mixed with part of m_2 (punctured)
- Joint decoding \rightarrow conventional decoding + backtrack decoding (using priors)

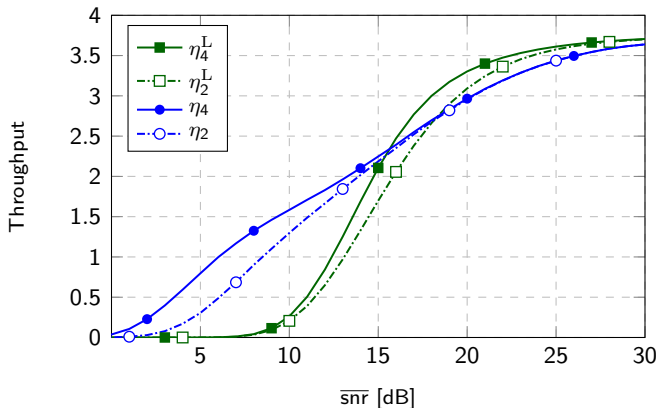
Higher rate, and mix at binary level \implies simpler encoders and decoders

Layer-coded HARQ



"Artificial" inclusion of systematic bits: if \hat{m}_2 is OK, m_2 is recovered, which provides the corresponding contribution to m_3

Example: 16-QAM, Turbo code, Rayleigh-fading, $R = 3.75$



- Gain for high throughput region (this is what we wanted!)
- Loss for low throughput (error propagation; should be combated with rate adaptation)

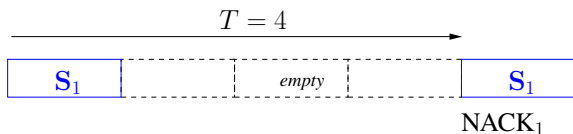
What if feedback not instantaneous ?

Management for T :

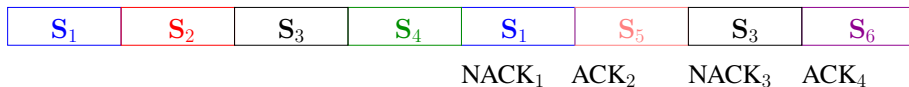
- Stop-and-Wait
- Parallel Stop-and-Wait/Selective Repeat

Management for T

STOP-AND-WAIT



PARALLEL/SELECTIVE-AND-REPEAT



Why $T \neq 1$?

- Decoding processing time at RX
- Framing : traffic for return channel
- Propagation time

Example: $T = 8$ in LTE

Non orthogonal HARQ; reducing the delay and improving the throughput

Another way of building multi-layer HARQ, with corresponding protocol.

- State of the Art ($T = 1$)
- Application to $T \neq 1$

State of the Art ($T = 1$)

Sending the superposition of two streams instead of one !

$$\mathbf{y} = \mathbf{x}_1 + \mathbf{x}_2 + \mathbf{w}$$

But superposition does not increase the capacity

$$R = R_1 + R_2 < \log_2(1 + P_1 + P_2) = \log_2(1 + P)$$

with P the transmit power.

However a way to be closer to the capacity, especially with retransmission (since ACK/NACK provides information)

Main Idea [Steiner06]:

- Frame 1: send two messages under superposition coding (SC), i.e., two layers with short power constraints P
- Frame 2: if one layer not decoded, send it again with full power P
- Frame 3: start with two new messages

Two contexts:

- Channel constant over each retransmission
- Channel time-varying at each retransmission

Additional works:

- Practical implementation of [Steiner06] with $P_1 = 0.8P$ [Assimi2009]
- CSI at the TX for relevant actions (SC or not with Markov Decision Process) [Jabi2015]
- At TCP level: flooding the TCP packet with hierarchical superposition coding [Zhang2009]

Application to $T \neq 1$

Idea To reduce the delay, send in advance (before receiving any ACK/NACK) redundant packets in superposition to standard parallel HARQ with low power (for minimizing the disturbance):

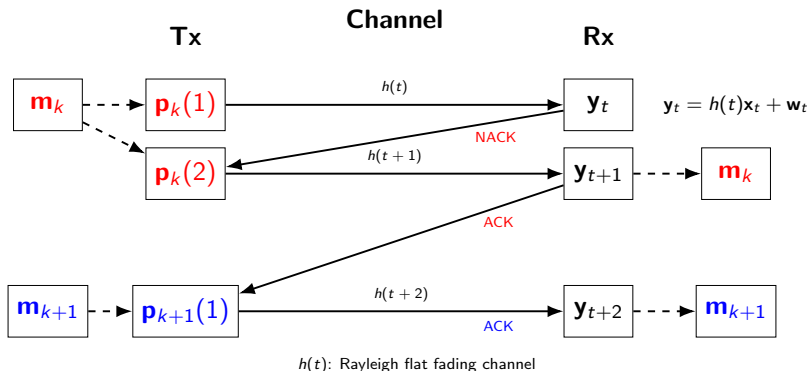
$$\begin{cases} \mathbf{S}_k(\ell), & \text{if no superposition,} \\ \sqrt{\alpha}\mathbf{S}_k(\ell) + \sqrt{1-\alpha}\mathbf{S}_{k'}(\ell'), & \text{if superposition.} \end{cases}$$

with k, k' the messages.

We have two layers :

- The first one is standard parallel HARQ
- The second one corresponds to superposed packets chosen as:
 1. $\mathbf{S}_{k'}(\ell')$ is not superposed if $\mathbf{m}_{k'}$ is in timeout or previously ACKed
 2. Superposed packet is the unsent packet of the lowest index ℓ' of the most recent message $\mathbf{m}_{k'}$, with $k' \neq k$
 3. If the transmitter already sent all the packets, superposed packet is with the lowest index ℓ' not previously sent in the second layer.
 4. No packet is superposed to a packet of the first layer that has $\ell = L$.

Hybrid ARQ (Automatic Repeat reQuest)



$\mathbf{p}_k(\ell)$: ℓ -th packet of message \mathbf{m}_k , $\ell \in \{1, \dots, C\}$

$\mathbf{p}_k(1) = \mathbf{p}_k(2)$

for CC-HARQ (Chase Combining)

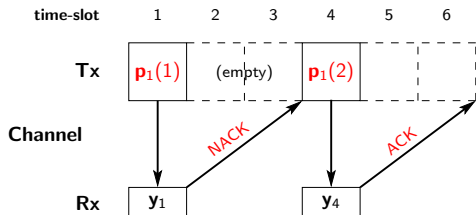
→ diversity gain

$\mathbf{p}_k(1) \neq \mathbf{p}_k(2)$

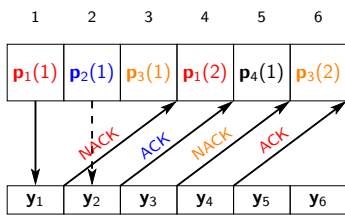
for IR-HARQ (Incremental Redundancy)

→ diversity + coding gain

HARQ with feedback delay ($T = 3$)



Stop-and-Wait



Parallel HARQ (Selective Repeat)

Why $T \neq 1$? ($T = 8$ in LTE)

- Decoding (processing) time at the receiver
- Framing: traffic for return channel
- Propagation time

Non-orthogonal transmission

Idea

- Superpose (re)transmitted packets to increase the throughput [Shamai08, Assimi09, Szczecinski14]

Objectives

- Low latency
- High reliability
- Large throughput

Why non-orthogonal transmission?

- Non-orthogonal transmission exploits the potential of MAC
- Other strategies usually require CSI at the transmitter [Kasper17]
 - time-sharing
 - rate adaptation

General idea

Send additional redundant packets using two layers

Before receiving the ACK/NACK feedback

Superposed to parallel HARQ

With low power

Layer 1: parallel HARQ **VERY important**

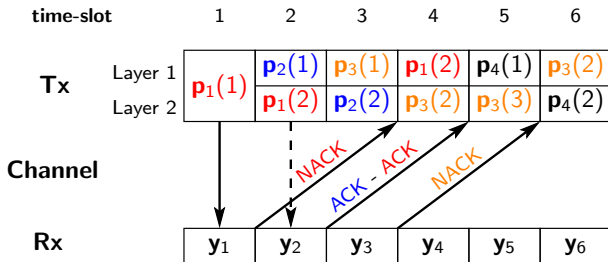
Layer 2: superposed packets

$$\mathbf{p}_k(\ell)$$

without superposition

$$\sqrt{\alpha}\mathbf{p}_k(\ell) + \sqrt{1-\alpha}\mathbf{p}_{k'}(\ell')$$

with superposition



Proposed protocol, $T = 3$

Transmitter

How do we choose the superposed redundant packets?

- Superpose packets of the most recent messages
→ Low latency
- Superpose unsent redundant packets
→ Transmit diversity
→ High reliability

		time-slot					
		1	2	3	4	5	6
Tx	Layer 1	$p_1(1)$	$p_2(1)$	$p_3(1)$	$p_1(2)$	$p_4(1)$	$p_3(2)$
	Layer 2		$p_1(2)$	$p_2(2)$	$p_3(2)$	$p_3(3)$	$p_4(2)$
		NACK	ACK ACK	NACK			

Proposed protocol, $T = 3$

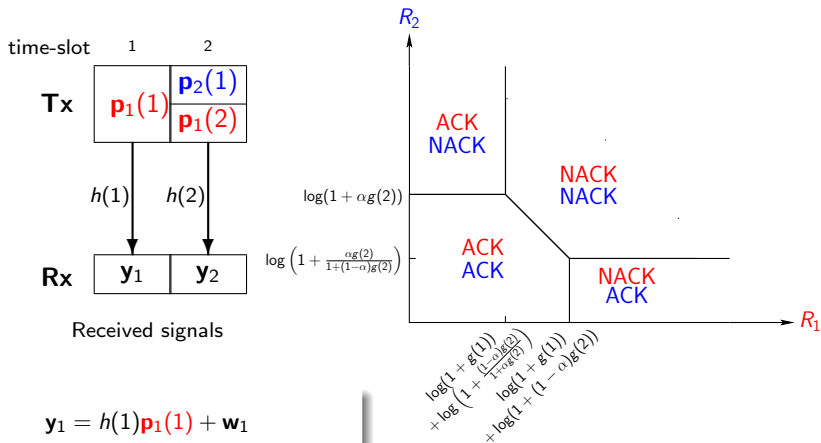
Low latency + High reliability → Large throughput

Decoding

Let \mathcal{M} be the set of messages that the receiver is attempting to decode at time-slot t .

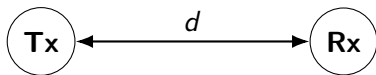
- If the receiver successfully decodes the subset $\mathcal{D} \subseteq \mathcal{M}$ and none of the messages in $\mathcal{M} \setminus \mathcal{D}$, we say that the decoder operates in the rate region $\mathcal{R}_{\mathcal{D}}$.
- The set \mathcal{D} , along with the rules of the transmit protocol, allows to obtain \mathcal{F}_t the set of ACK/NACK.
- In order to characterize the decoding outcome, we
 1. evaluate the rate region $\mathcal{R}_{\mathcal{D}}$ for every possible $\mathcal{D} \subseteq \mathcal{M}$, by checking the corresponding rate inequalities
 2. determine, on the basis of the available observations, the operating rate region $\mathcal{R}_{\mathcal{D}}$ of the receiver.

Performance with capacity-achieving codes

Rate regions at $t = 2$ [ElGamal12]

$$g(t) = |h(t)|^2$$

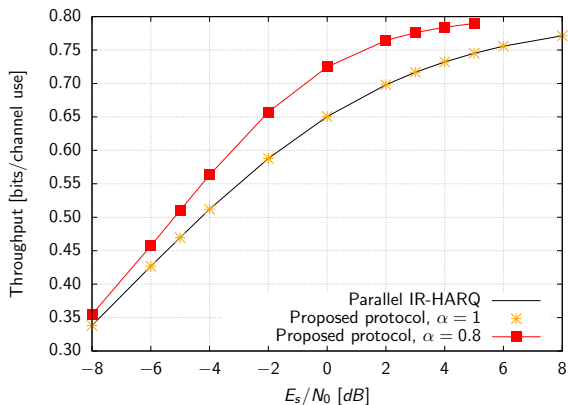
Setup for numerical evaluation



Distance between the transmitter and the receiver
 $d = 15u$ where u is a unit of distance

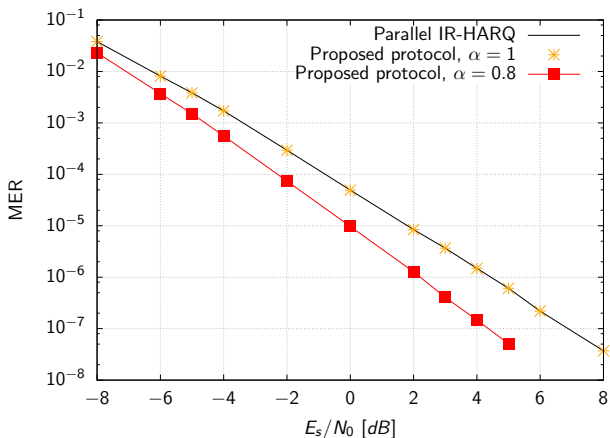
- **Variance** : $\sigma^2 = \left(\frac{c}{d^2}\right)^2$ where c is a constant, fixed as $c = 400u^2$
- **HARQ protocol** : IR-HARQ with $C = 4$ and $R = 0.8$
- **Feedback delay** : $T = 3$ time-slots
- **Transmit energy** : E_s per symbol

Throughput using capacity-achieving codes



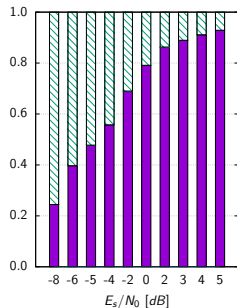
1dB to 2.5dB gain at moderate SNR, Much more for high SNR
10% throughput gain at 0dB

Message Error Rate using capacity-achieving codes



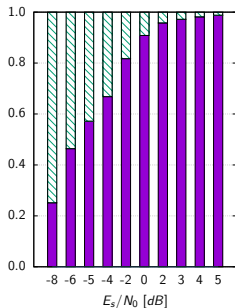
Additional diversity gain due to multi-layer transmission

Latency using capacity-achieving codes



Parallel IR-HARQ

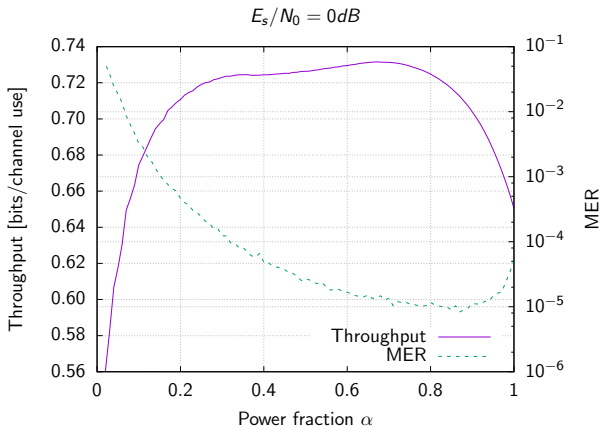
1-3 time-slots
4-10 time-slots



Proposed protocol

1-3 time-slots
4-10 time-slots

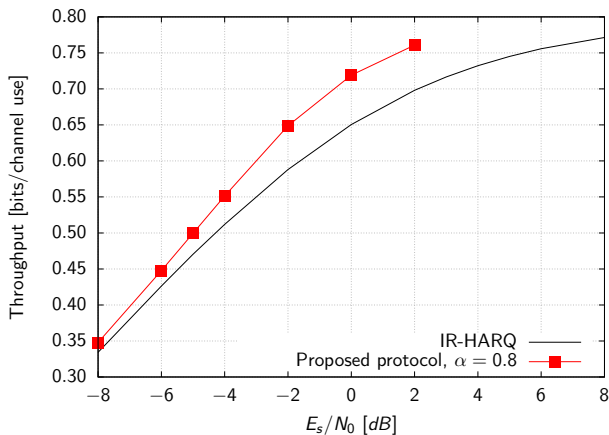
More packets are served with small delays (< 4 time-slots)

Numerical optimization of α 

$\alpha = 0.7$ provides the best performance at $0dB$
 α can be numerically optimized for each SNR

Proposed protocol in comparison to 3GPP LTE

Throughput using $C = 4$, $T = 8$ and capacity-achieving codes

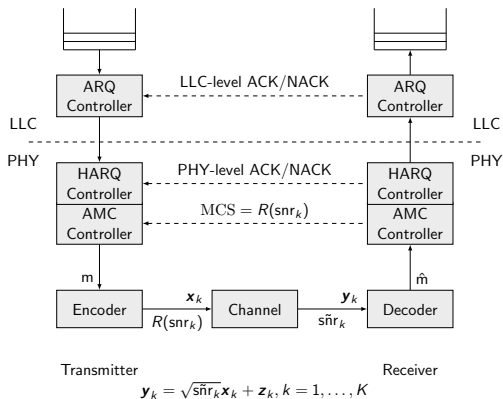


Part 5 : HARQ and AMC; Friends or Foes?

- 5.1 Model again; the source of errors
- 5.2 HARQ on top of AMC; problems and remedies
- 5.3 Connecting L-HARQ with AMC

Previously : the rate was fixed, but now, we take into account the fact that (average) CSI knowledge allows AMC: what happens when combined with HARQ ?

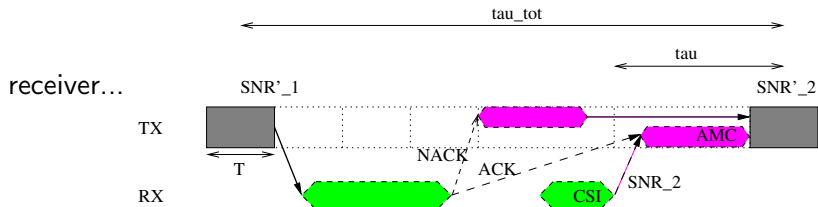
Model: AMC+HARQ, saturated buffer



- Only PHY throughput counts: LLC-level ARQ removes all residual errors from PHY
- Modulation and coding set (MCS) is decided by the receiver (using measured CSI)
- Measured CSI ($s\tilde{n}r$) is delayed with respect to the actual CSI ($s\tilde{n}r$)

Decoding Errors due to the delayed CSI (Doppler)

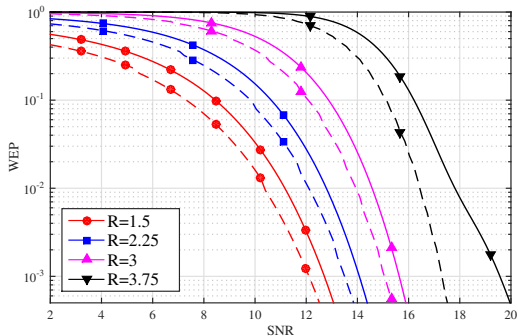
2 different SNR's perceived at the transmitter: the "average" on which AMC is chosen, and the instantaneous (but outdated) one coming from



Assumptions

- Propagation time is (often) negligible
- Processing time is non-negligible for decoding, CSI acquisition, encoding
- $\tau_{tot} f_D \gg 1$ (snr'_1 and snr'_2 are independent)
- $\tau_{tot} f_D > 0$ (snr_2 and snr'_2 are correlated)
- $T f_D \approx 0$ (no channel variation when receiving)

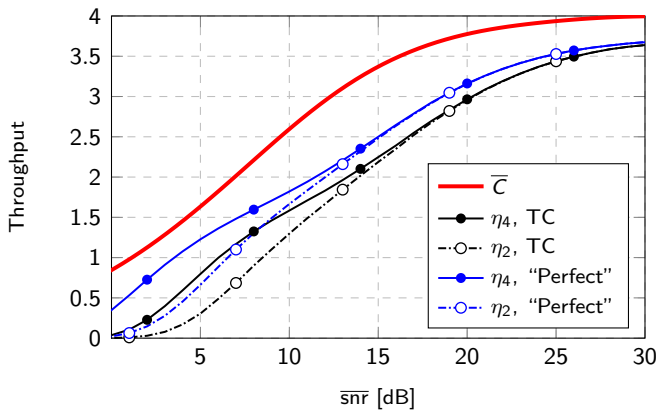
Example of PER curves; 16-QAM; Doppler $f_D\tau = 0.05$



- Theoretical and practical curves are similar: Turbo-C (solid) and PerfectC (dashed)
- In practice: fix decoding threshold, PER_{th} and select MCS using snr
- For example: $PER_{th} = 0.1$, $R([8.5\text{dB}, 10.5\text{dB}]) = 1.5$, $R([10.5\text{dB}, 13\text{dB}]) = 2.25$, etc.

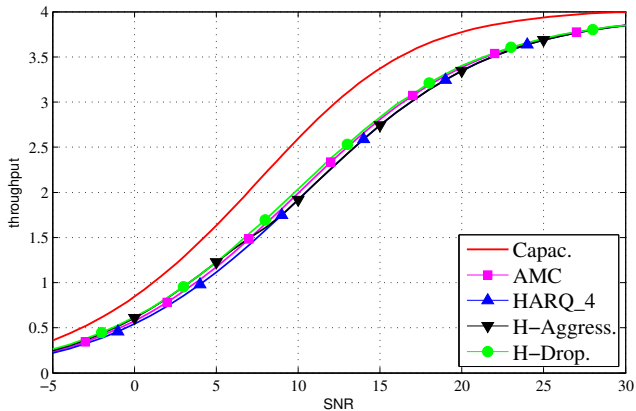
Problem : curves should be "observed" (measured) for each possible receiver
: decoding time has an impact...

Reminder: How much we gain with HARQ (no AMC)

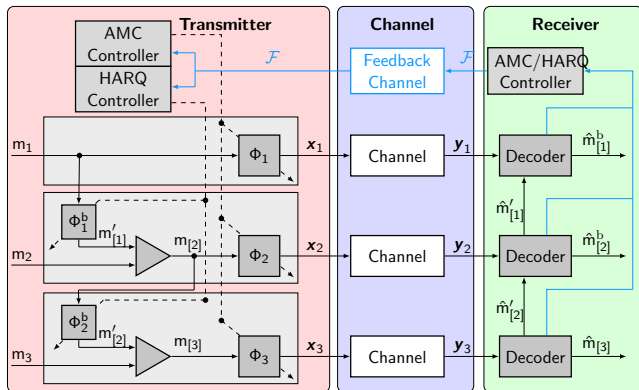


- Throughput improved in low SNR
- No gain for high nominal rate, i.e., in high SNR

How much we loose (!) using HARQ + AMC



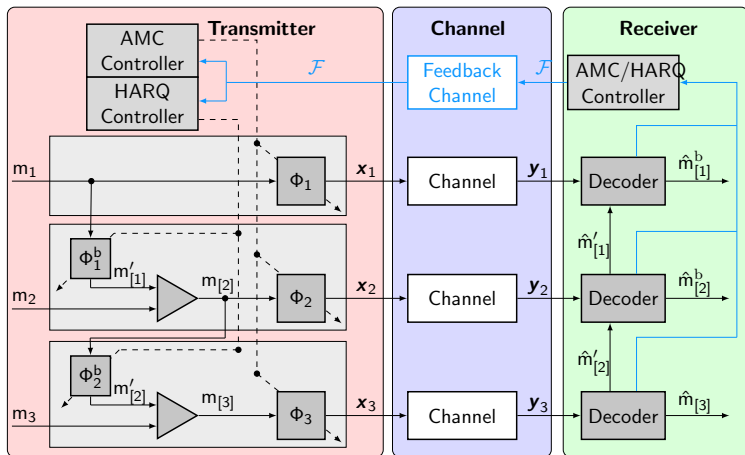
- Throughput degradation in high SNR due to HARQ
- Source of the problem: i) first round rate $R_1 = R(\text{snr}_1)$; ii) after NACK, second round's reward is only R_1 ; iii) in AMC the reward might be $R_2 = R(\text{snr}_2) > R_1$
- Patching: if $\text{snr}_2 > \text{snr}_1$, abandon HARQ and use AMC (packet dropping)

AMC+L-HARQ: decoding example, $k = 3$ 

Φ 's are controlled by AMC, Φ^b 's controlled by HARQ

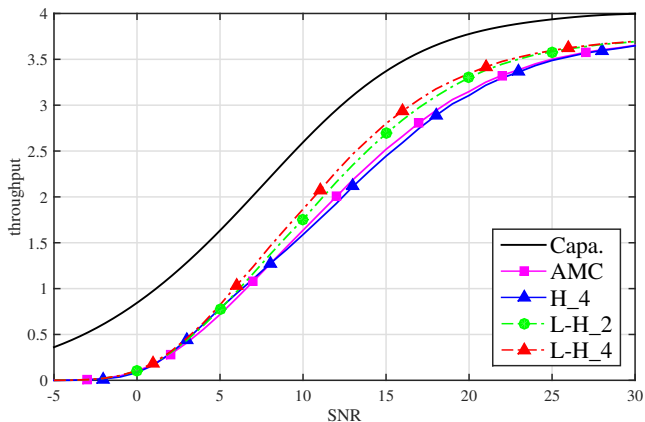
- $\text{NACK}_1, \text{NACK}_2, \text{ACK}_3 \rightarrow \hat{m}^b_{[3]}$ and $\hat{m}^b_{[2]}$ are error-free
- Backtrack decoding: using $\hat{m}'_{[2]}$, decoder #2 produces $\hat{m}^b_{[2]}$ and $\hat{m}'_{[1]}$, which are error-free
- Backtrack decoding: using $\hat{m}'_{[1]}$, decoder #1 produces $\hat{m}^b_{[1]}$ which is error-free

AMC+L-HARQ: Decoupled control



- AMC round k : Channel encoder Φ_k (MCS) adapts to fresh CSI (measured at round k)
- HARQ round k : Puncturer Φ_{k-1}^b adapts to old CSI (from the round $k-1$)
- Joint optimization of rates not needed
- Knowledge of the channel model not needed for optimization (of the rates)

Numerical example: Turbo C, 16QAM, Rayleigh, $\tau f_D = 0.05$ and $R \in \{1.5, 2.25, 3, 3.75\}$



- The adaptation does not depend on the channel model (emphasized again)
- HARQ improves with number of rounds (that's what we wanted!)
- Gains $\sim 3\text{dB}$ for high rates,

Part 6 : Extensions and wrap up

Content :

- cooperative communications
- conclusions on theoretical and practical issues

Introduction

Interaction between Relaying and HARQ:

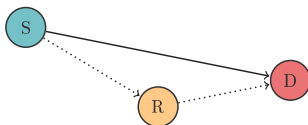
- Both techniques applied solely will bring improvement;
- What improvement will bring if these two techniques are applied together?
- What is the best way of combining them?

Reference literature

- Combination of these two techniques in literature:
 - Energy efficiency is studied in [Stanojev, 2009], and from the perspective of information theory is studied in [Falavarjani, 2010];
 - The interaction is mostly studied via deterministic protocols [Krikidis, 2007]; We focus on both: deterministic and probabilistic protocols;
 - The Relay is mostly considered in Decode-and-Forward (DCF) mode; We focus more on the Demodulate-and-Forward (DMF) mode.
- For theoretical analysis we focus on Finite State Markov Chain (FSMC).

System model

- Example scenario:
 - Source-Relay-Destination network;
 - ARQ mechanism (stop-and-wait policy);
 - All the nodes listen to control messages (ACK/NACK) issued by D.

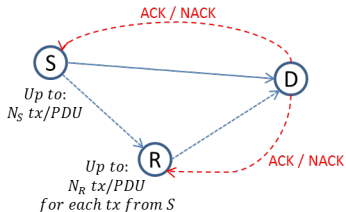


- Relay mode:
 - Decode-and-Forward (DCF) - Relay always forward the correct copy.
 - Demodulate-and-Forward (DMF) - demodulation errors of R are taken into account when evaluating likelihood function at the decoder:

$$p(y_{RD,n}|c_{n,i}) = p(y_{RD,n}|D_R=0, c_{n,i}) p(D_R=0|c_{n,i}) + p(y_{RD,n}|D_R=1, c_{n,i}) p(D_R=1|c_{n,i})$$

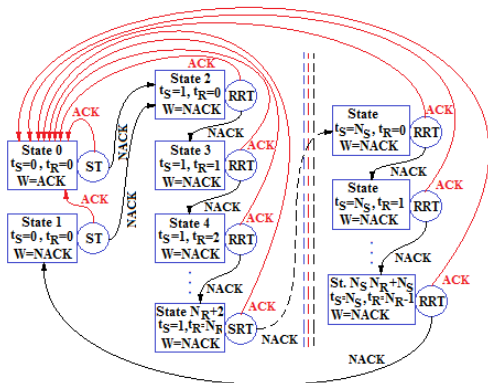
The deterministic protocol, DMF mode

- The example protocol:



- Finite State Machine (FSM):
 - Systematic way for analyzing protocols;
 - FSM enters a state in each time-slot;
 - The state determines the action that is going to be taken during the time-slot;
 - The outcome of the action determines the transition to the next state.

From FSM to FSMC, DMF



- Monte Carlo simulation for evaluation of:
 - $\pi_{[1,0]}$ - probability of NACK on the channel S-D;
 - $\pi_{[0,1]}$ - probability of NACK on the channel R-D;
 - $\pi_{[A,B]}$ - prob. of NACK combining A cop. from S and B cop. from R.

Probability transition matrices, DMF

$$P_I = \begin{pmatrix} 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 & \cdots & 0 & \cdots \\ 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 & \cdots & 0 & \cdots \\ 1 - \pi_{[0,1]} & 0 & 0 & \pi_{[0,1]} & \cdots & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \\ 1 - \pi_{[1,0]} & 0 & 0 & 0 & \cdots & \pi_{[1,0]} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \\ 1 - \pi_{[0,1]} & \pi_{[0,1]} & 0 & 0 & \cdots & 0 & \cdots \end{pmatrix}$$

$$P_{II} = \begin{pmatrix} 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 & \cdots & 0 & \cdots \\ 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 & \cdots & 0 & \cdots \\ 1 - \pi_{[1,1]} & 0 & 0 & \pi_{[1,1]} & \cdots & 0 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \\ 1 - \pi_{[1,N_R]} & 0 & 0 & 0 & \cdots & \pi_{[1,N_R]} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots & \ddots \\ 1 - \pi_{[N_S, N_S N_R]} & \pi_{[N_S, N_S N_R]} & 0 & 0 & \cdots & 0 & \cdots \end{pmatrix}$$

Performance evaluation using FSMC

- Performance metrics:
 - PDU error rate (PER) - the proportion of PDUs that were transmitted but never ACK-ed by D;
 - \bar{T} - average number of transmissions per PDU;
 - Goodput (G) - the number of successfully delivered information PDU's per unit of time.
- Performance analysis using FSMC representation:
 - We evaluate the steady state vector \mathbf{p} from matrix P_I or P_{II} ;
 - We obtain the steady state probabilities of the initial states p_0 and p_1 ;
 - The performance metrics can be obtained as:

$$PER = \frac{p_1}{p_0 + p_1}, \quad \bar{T} = \frac{1}{p_0 + p_1}$$

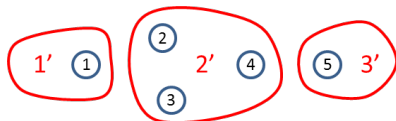
$$G = R_c \cdot \frac{1 - PER}{\bar{T}} \left[\frac{PDUs}{tu} \right] = R_c \cdot p_0 \left[\frac{PDUs}{tu} \right]$$

Accurate performance evaluation.... but can become computationally expensive

- As the protocol gets more sophisticated, the FSMC analysis becomes more complex:
 - Increasing the number of nodes or the number of transmissions, the number of states increases very quickly;
 - Switching the Relay from DMF mode to DCF mode, the number of states increases also quickly.
- Resulting number of nodes can quickly become much larger than 100, hence:
 - can we reduce the size of the FSMC while keeping PER, \bar{T} and G , untouched ? (equivalent to keep State 0 and State 1 untouched);
 - Since each state is associated with an action, it is more straightforward to aggregate states with the same actions.

State aggregation on the FSMC

- Let us consider the following example:



- If l is a new state resulting from the aggregation of the set of states \mathcal{I} , then the steady state probability of being in state l is:

$$z_l = \sum_{i \in \mathcal{I}} p_i.$$

- The transition probabilities between the aggregated states can be evaluated as:

$$Z_{IJ} = \frac{\sum_{i \in \mathcal{I}} p_i \left(\sum_{j \in \mathcal{J}} P_{ij} \right)}{\sum_{i \in \mathcal{I}} p_i}.$$

State aggregation: simplified FSMC, DMF

- The simplified transition matrix contains only four states:

$$Z = \begin{bmatrix} 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 \\ 1 - \pi_{[1,0]} & 0 & \pi_{[1,0]} & 0 \\ 1 - \pi_{[RF]} & \gamma \cdot \beta \pi_{[RF]} & (1 - \gamma) \pi_{[RF]} & \gamma (1 - \beta) \pi_{[RF]} \\ 1 - \pi_{[SF]} & 0 & \pi_{[SF]} & 0 \end{bmatrix}.$$

where parameters $\pi_{[RF]}$, $\pi_{[SF]}$, γ and β link the original transition matrix with the simplified one, and can be obtained from the state aggregation procedure;

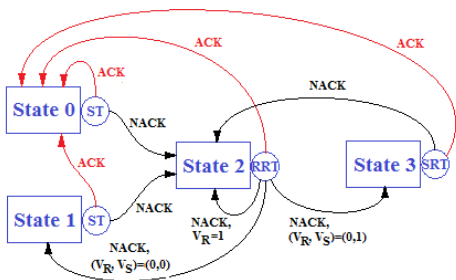
- The idea of state aggregation can be extended similarly to the case of DCF mode.

Protocol associated with the simplified FSMC

- Aggregation of states:
 - The actions remain the same;
 - Some transitions now will become probabilistic;
 - If we define:
 - γ - the probability that R is not allowed to retransmit one more time after it failed previously;
 - β - the probability that S is not allowed to retransmit one more time after R failed and is not allowed to retransmit anymore.
 - We can associate the simplified transition matrix Z with a FSM and a protocol.

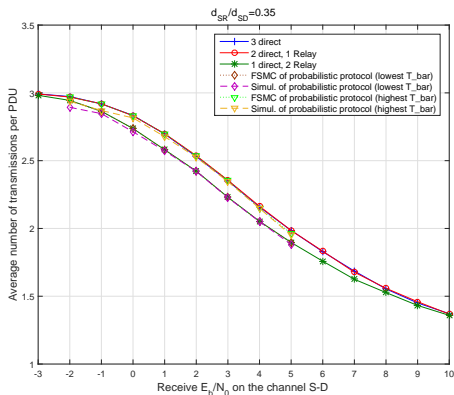
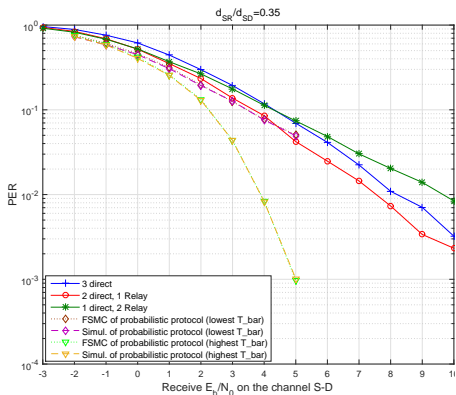
The probabilistic protocol: FSM at the transmitter

- Definition of the probabilistic protocol:
 - The protocol starts either from State 0 or from State 1;
 - If NACK from D: the first retransmission comes from R;
- If R is retransmitting, the next action is chosen by realization of two random parameters V_S and V_R :
 - R retransmits with probability $(1 - \gamma)$;
 - S retransmits with probability $(\gamma(1 - \beta))$;
 - Neither S or R are allowed to retransmit, with probability $\gamma \cdot \beta$. The PDU is lost.



Comparison with a reference protocol, type II decoder

- Comparison with a referent deterministic protocol:
 - Comparison in PER and \bar{T} ;



In summary

- HARQ is "yet another" way of adapting the communication protocol to the actual channel values, therefore ...
 - the compatibility with other ingredients of the protocol has to be checked
 - and some adaptation has to be implemented
- but these adaptations also open new possibilities, with improved performance... or not !
- Clearly, non orthogonal superposition instead of orthogonal retransmission has a great potential of improvement...

References (1)

- pp. 31-33
 - G. Caire and D. Tuninetti, "The throughput of hybrid-ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971-1988, July 2001.
 - P. Larsson, L. K. Rasmussen, and M. Skoglund, "Throughput analysis of ARQ schemes in Gaussian block fading channels," *IEEE Trans. Commun.*, vol. 62, no. 7, pp. 2569-2588, Jul. 2014.
 - M. Jabi, A. Benyouss, M. Le Treust, E. Pierre-Doray, and L. Szczecinski, "Adaptive Cross-Packet HARQ," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2022-2035, May 2017.

- pp. 34-37
 - L. Szczecinski, S. Khosravirad, P. Duhamel, and M. Rahman, "Rate Allocation and Adaptation for Incremental Redundancy Truncated HARQ," *IEEE Trans. Commun.*, vol. 61, no. 6, pp. 2580-2590, Jun. 2013.
 - S. Pfletschinger, D. Declercq, and M. Navarro, "Adaptive HARQ with non-binary repetition coding," *IEEE Trans. Wireless Commun.*, vol. 13, no. 8, pp. 4193-4204, Aug. 2014.
 - M. Jabi, L. Szczecinski, M. Benjillali, and F. Labeau, "Outage Minimization via Power Adaptation and Allocation for Truncated HARQ," *IEEE Trans. Commun.*, vol. 63, no. 3, pp. 711-723, Mar. 2015.
 - M. Jabi, M. Benjillali, L. Szczecinski, and F. Labeau, "Energy Efficiency of Adaptive HARQ," *IEEE Trans. Commun.*, vol. 64, no. 2, pp. 818-831, Feb. 2016.
 - W. Su, S. Lee, D. Pados, and J. Matyjas, "Optimal power assignment for minimizing the average total transmission power in hybrid-ARQ Rayleigh fading links," *IEEE Trans. Commun.*, vol. 59, no. 7, pp. 1867-1877, Jul. 2011.
 - T. Chaitanya and E. Larsson, "Optimal power allocation for hybrid ARQ with chase combining in i.i.d. Rayleigh fading channels," *IEEE Trans. Commun.*, vol. 61, no. 5, pp. 1835-1846, May 2013.

- pp. 38-42
 - M. Jabi, A. Benyouss, M. Le Treust, E. Pierre-Doray, and L. Szczecinski, "Adaptive Cross-Packet HARQ," *IEEE Trans. Commun.*, vol. 65, no. 5, pp. 2022-2035, May 2017.
 - M. Jabi, A. El Hamss, L. Szczecinski, and P. Piantanida, "Multi-Packet Hybrid ARQ: Closing Gap to Ergodic Capacity," *IEEE Trans. Commun.*, vol. 63, no. 12, pp. 5191-5205, Dec. 2015.

References (2)

- pp.45-47
 - M. Jabi, E. Pierre-Doray, L. Szczecinski, and M. Benjillali, "How to Boost the Throughput of HARQ with Off-the- Shelf Codes," *IEEE Trans. Commun.*, vol. 65, no. 6, pp. 2319-2331, June 2017.
 - P. Popovski, "Delayed channel state information: Incremental redundancy with backtrack retransmission," in *IEEE Inter. Conf. Comm. (ICC)*, June 2014, pp. 2045-2051.
- pp. 55-65
 - A. khreis, Ph. Ciblat, F. Bassi, and P. Duhamel. "Multi-Packet HARQ with Delayed Feedback." In *International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, Bologne, Italy, September 2018.
- pp. 67-73
 - Q. Liu, S. Zhou, and G. B. Giannakis, "Cross-layer combining of adaptive modulation and coding with truncated ARQ over wireless links," *IEEE Trans. Wireless Commun.*, vol. 3, no. 5, pp. 1746-1755, Sep. 2004.
 - R. Sassioui, M. Jabi, L. Szczecinski, L. B. Le, M. Benjillali, and B. Pelletier, "HARQ and AMC: Friends or Foes ?," *IEEE Trans. Commun.*, vol. 65, no. 2, pp. 635-650, Feb. 2017.
 - M. Jabi, L. Szczecinski, M. Benjillali, A. Benyouss, and B. Pelletier, "AMC and HARQ: How to Increase the throughput," *IEEE Trans. Commun.*, vol. 66, no. 6, pp. 3136-3150, July 2018.
- pp. 77-87
 - A. Vanyan, F. Bassi, A. Herry, and P. Duhamel. "Coding, diversity and ARQ in fading channels: a case-study performance comparison". In *24th Annual International Symposium on Personal, Indoor, and Mobile Radio Communications (PIMRC)*, pages 175, Londres, United Kingdom, September 2013. doi: 10.1109/pimrc.2013.6666377.
 - F. Maliqi, P. Duhamel, F. Bassi, and I. Limani. "Simplified Analysis of HARQ Cooperative Networks Using Finite-State Markov Chains". In *European Signal Processing Conference (EUSIPCO)*, Kos, Greece, August 2017. *Eurasip*. doi: 10.23919/eusipco.2017.8081561.
 - F. Maliqi, F. Bassi, P. Duhamel, and I. Limani. "A probabilistic HARQ protocol for Demodulate-and-Forward (DMF) relaying network". *IEEE Transactions on Wireless Communications*, 2019. doi: 10.1109/TWC.2019.2894642.