NOLLA: Non-Linear Outer Loop Link Adaptation for Enhancing Wireless Link Transmission

Lingrui Zhu*, Carsten Bockelmann*, Thorsten Schier†, Salah Eddine Hajri†, Armin Dekorsy*
*Department of Communications Engineering, University of Bremen, 28359, Bremen, Germany
Email: {zhu, bockelmann, dekorsy}@ant.uni-bremen.de
†Huawei Technologies Sweden AB, 16903 Solna, Sweden
Email: {thorsten.schier, salah.eddine.hajri}@huawei.com

***Abstract*—Modern wireless systems require strict key performance indicators (KPIs), such as very high reliability and throughput. Link adaptation (LA) is a core technology used to achieve these targets, with outer loop link adaptation (OLLA) being the most commonly used algorithm due to its feasibility and simplicity. OLLA uses a term called backoff factor to correct the signal to interference and noise ratio (SINR) estimate mapped from channel state information (CSI) to obtain an effective SINR. Based on the effective SINR, modulation and coding scheme (MCS) will be selected. OLLA adjusts the backoff factor with respect to hybrid automatic repeat request (HARQ) in a linear manner. However, this leads to effective SINR fluctuation and quite often results in overestimation. Hence, the performance of the system will be degraded. In this work, we propose a novel algorithm which introduces an adaptive adjustment step size for the backoff factor using an exponentially decaying factor to alleviate this issue. Since the backoff factor is not adjusted in a linear manner like OLLA, we call the proposed novel algorithm non-linear outer loop link adaptation (NOLLA). NOLLA can be regarded as an extension of OLLA that retains low complexity and high feasibility, providing the possibility to improve the link transmission in an uncomplicated way. Numerical evaluations demonstrate that NOLLA achieves higher reliability and throughput in scenarios with and without interference.**

***Index Terms*—outer loop link adaptation, adaptive step size.**

I. Introduction

As technologies advance, emerging services, for example, autonomous vehicles [1] and extended reality (XR) [2], demand wireless communication systems to meet increasingly stringent requirements in terms of reliability and throughput. Link adaptation (LA) [3] is a key technology in modern wireless communication systems, including cellular networks, wireless LANs and satellite communication systems to find a reasonable balance between these two critical key performance indicators (KPIs). With LA, link transmissions are able to select the proper modulation and coding schemes (MCS) according to the channel state information (CSI) in a flexible manner. Inner loop link adaptation (ILLA) and outer loop link adaptation (OLLA) [4] are the two most common methods used in wireless communication systems. ILLA just uses the signal-to-interference-plus-noise ratio (SINR) mapped from CSI to select corresponding MCS, while OLLA applies a backoff factor to correct the SINR estimated from CSI to obtain an effective SINR for MCS selection. The backoff factor is adjusted with respect to hybrid automatic repeat request (HARQ) feedback.

978-1-6654-6483-3/2/ $31.00 ©2023 IEEE

In recent years, many efforts have been made to enhance the performance of LA algorithms. In [5], logistic regression has been used to model the block error rate (BLER) with the aim of maximizing the throughputs. Link adaptation for 5G vehicular networks has been enhanced by introducing a Markov chain to predict latency, BLER and throughput in [6]. Moreover, machine learning methods are also employed to improve the performance of LA approaches. Authors of [7] utilize autoencoder (AE) and support vector machine (SVM) to select MCS. Reinforcement learning based LA methods have been proposed to achieve higher throughputs in [8]. Thompson sampling has been introduced to learn MCS success probabilities for LA in [9]. The problem for learning based methods is that data driven approaches always require high computational complexity.

In our work, a new algorithm called non-linear outer loop link adaptation (NOLLA) is proposed. When adjusting the backoff factor for OLLA, the step size is fixed. Therefore, OLLA is prone to overestimate the effective SINR and it may result in heavy fluctuation. As a result, the throughput and reliability will be influenced. However, in NOLLA, an exponential decaying factor is introduced to make the step-size adjustable. This decaying factor is updated based on the sequence of HARQ feedbacks.

Our contributions can be summarized as following:

- We analyze OLLA and propose NOLLA which introduces an adaptive step size to make backoff factor adjustment more stable.
- We derive the mathematical description of our newly introduced parameters in NOLLA and explain the parameter choice.
- We numerically evaluate the performance of OLLA and NOLLA in terms of throughput and reliability in different scenarios.

The rest of this paper is structured as follows. In section II, our system model and problem formulation are presented. Algorithms, including OLLA and NOLLA, as well as related derivations are introduced in section III. The performances of OLLA and NOLLA are numerically evaluated in section IV. Finally, section V provides the conclusion of this paper.

II. System Model and Problem Formulation

A. System Model

Consider a downlink transmission from a gNodeB (gNB) to a user equipment (UE), as illustrated in Fig. 1. Prior to transmitting data, the gNB sends a channel state information reference signal (CSI-RS) to the UE via the physical downlink shared channel (PDSCH). Using CSI-RS, the UE estimates the SINR, which is mapped to a channel quality indicator (CQI) index in a CSI report sent back to the gNB via physical uplink control channel (PUCCH). Based on the CQI index, the gNB estimates the SINR and selects the appropriate MCS for the upcoming transmission. Upon receiving the data, the UE performs a cyclic redundancy check (CRC) to verify the success of the transmission and sends the corresponding hybrid automatic repeat request (HARQ) feedback to the gNB. If data is received successfully, feedback ACK will be sent back. Otherwise, NACK will be sent.

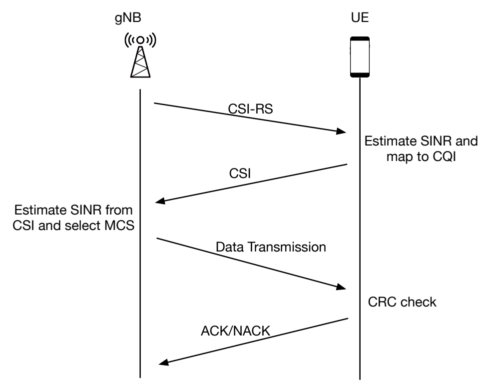

Fig. 1: The downlink transmission with CSI feedback and link adaptation.

B. Problem Formulation

As introduced in II-A, the selection of MCS is crucial in LA algorithms. It determines the trade-off between achieving a high throughput and keeping BLER within the system requirements. Thus, the selection of MCS can be formulated as an optimization problem:

$$\begin{aligned} \max_{m \in \mathcal{M}} \quad & {}_m \cdot [1 - {}_m(\hat{})] \\ \text{s.t.} \quad & {}_m(\hat{}) \leq \text{BLER}_{\text{target}} \end{aligned} \tag{1}$$

where $\mathcal{M} = \{1; 2; :::; N_{\text{MCS}}\}$ indicates the available MCS for the transmission system and N_{MCS} is the number of available MCSs. ^ denotes estimated SINR. The function ${}_m(\cdot)$ models BLER for MCS m and the logistic regression model [5] is employed to model BLER in this work. $\text{BLER}_{\text{target}}$ is the target BLER. The throughput of a link transmission with MCS m can be calculated applying the equation shown below:

$${}_m = \frac{\text{TBS}_m}{f \cdot \ t} = \frac{N_{\text{RE;data}} \cdot R_m \cdot Q_m}{f \cdot \ t} \tag{2}$$

TBS_m denotes the transport block size for MCS m. f and t are transmission bandwidth and duration, respectively. MCS m can be broken down in to code rate R_m and modulation order Q_m. The number of resource elements for data transmission in the transport block is represented by $N_{\text{RE;data}}$, which can be calculated as:

$$N_{\text{RE;data}} = N_{\text{sc}} \cdot N_{\text{symbols}} - N_{\text{DMRS}} - N_{\text{oh}} \tag{3}$$

Number of subcarriers and symbols of a single transport block is denoted as N_{sc} and N_{symbols} correspondingly. In addition, N_{DMRS} and N_{oh} represent the amount of demodulation reference signal (DMRS) and overhead in the transport block as depicted in Fig. 2.

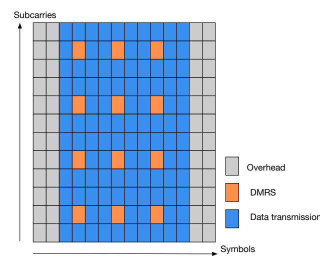

Fig. 2: An example of resource blocks and resource elements. X-axis indicates time symbols and Y-axis indicates frequency subcarriers. Each grid is a resource element.

III. Link Adaptation Algorithms

A. OLLA

OLLA [4] uses a backoff factor ${}_{\text{OLLA}}$ to adjust the SINR estimate as follows:

$$\hat{}_{\text{eff}} = \hat{}_{\text{CQI}} - {}_{\text{OLLA}} \tag{4}$$

where $\hat{}_{\text{CQI}}$ is the SINR estimate based on CQI and $\hat{}_{eff}$ is effective SINR estimate from OLLA. The backoff factor will be adjusted according to HARQ feedback and it can be calculated using the following equation at transmission time interval (TTI) t:

$${}_{\text{OLLA}}[t] = \begin{cases} {}_{\text{OLLA}}[t-1] - {}_{down}; & \text{if } e[t] = 0 \\ {}_{\text{OLLA}}[t-1] + {}_{up}; & \text{if } e[t] = 1 \end{cases} \tag{5}$$

where $e[t]$ is the error indicator function. It equals to 0 when the packet has been successfully received and 1 otherwise. ${}_{\text{up}}$ and ${}_{\text{down}}$ are SINR adjustment step sizes. The intention of the adjustment of ${}_{\text{OLLA}}$ is to increase the backoff factor when transmission errors happen. As a result, the effective SINR $\hat{}_{\text{eff}}$ will be underestimated to take a more conservative MCS selection policy.

The ratio of ${}_{\text{up}}$ to ${}_{\text{down}}$ is also given in [4]:

$$\frac{{}_{\text{up}}}{{}_{\text{down}}} = \frac{1 - \text{BLER}_{\text{target}}}{\text{BLER}_{\text{target}}} \tag{6}$$

The OLLA algorithm is listed in Algorithm 1 and it is not difficult to find that ${}_{\text{up}}$ is always way greater than ${}_{\text{down}}$. For example, if $\text{BLER}_{\text{target}}$ is 0.1 and ${}_{\text{down}}$ is initialized to 0.1, ${}_{\text{up}}$ should be 0.9. The philosophy of OLLA is to increase the effective SINR estimate $\hat{}_{\text{eff}}$ with small steps carefully when

Algorithm 1 Outer Loop Link Adaptation

Initialize: Define $_{\mathrm{init}}$ and $\mathrm{BLER}_{\mathrm{target}}$.
$_{\mathrm{OLLA}} = 0$, $_{\mathrm{down}} = {}_{\mathrm{init}}$, $_{\mathrm{up}} = \frac{1-\mathrm{BLER}_{\mathrm{target}}}{\mathrm{BLER}_{\mathrm{target}}}\ {}_{\mathrm{init}}$
for Time slots $t = 1$ to N_{TTI} **do**
  if $t \mod T_{\mathrm{CQI}} = 0$ **then** . T_{CQI} denotes CQI interval
    Update estimated SINR $\hat{}_{\mathrm{CQI}}$ based on new CQI.
  end if
  if ACK received **then**
    Update: $_{\mathrm{OLLA}}[t] = {}_{\mathrm{OLLA}}[t \quad 1] \quad {}_{\mathrm{down}}$
  else if NACK received **then**
    Update: $_{\mathrm{OLLA}}[t] = {}_{\mathrm{OLLA}}[t \quad 1] + {}_{\mathrm{up}}$
  end if
  calculate effective SINR: $\hat{}_{\mathrm{eff}} = \hat{}_{\mathrm{CQI}} \quad {}_{\mathrm{OLLA}}[t]$
  select MCS according to $\hat{}_{\mathrm{eff}}$ based on Eq.(1).
end for

ACK is received and reduce $\hat{}_{\mathrm{eff}}$ to a large degree when NACK is received as depicted in Fig. 3.

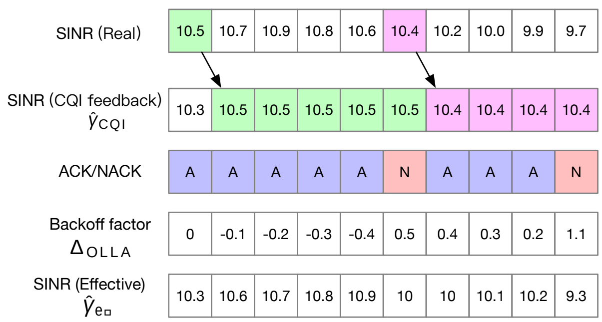

Fig. 3: An Example of SINR estimation (in dB) of OLLA. CQIs are transmitted every 5 TTIs ($T_{\mathrm{CQI}} = 5$). $\hat{}_{\mathrm{eff}}$ grows slowly when receiving ACKs but drops dramatically when NACK is received.

B. NOLLA

The procedure repeated by OLLA can be regarded as a "Climb and Fall" process. At first, it increases $\hat{}_{\mathrm{eff}}$ with several consecutive ACKs, as a result, the MCS with higher order will be selected. The higher the MCS order, the greater the likelihood of receiving a NACK. When HARQ feedback is NACK, $\hat{}_{\mathrm{eff}}$ will be decreased dramatically. To reduce BLER, NOLLA lengthens the "Climb" phase and decreases the frequency of the "Fall" phase. Instead of adjusting the backoff factor $_{\mathrm{OLLA}}$ with constant step sizes $_{\mathrm{up}}$ and $_{\mathrm{down}}$ as given in (5), an exponential decaying factor is introduced in NOLLA and the adjustment of $_{\mathrm{OLLA}}$ can be written as:

$$_{\mathrm{OLLA}}[t] = \begin{cases} {}_{\mathrm{OLLA}}[t-1] - {}_{\mathrm{down}}^{M_{\mathrm{A}}[t]} \cdot {}_{\mathrm{down}}; & \text{if ACK} \\ {}_{\mathrm{OLLA}}[t-1] + {}_{\mathrm{up}}^{M_{\mathrm{N}}[t]} \cdot {}_{\mathrm{up}}; & \text{if NACK} \end{cases}; \quad (7)$$

where $_{\mathrm{up}}$ and $_{\mathrm{down}} \in [0; 1]$ are exponential bases for decaying factors. M_{A} and M_{N} are called momentum index of ACK and NACK, respectively, indicating the number of consecutive ACKs/NACKs received, as the example depicted in Fig. 4.

Notice that, it is necessary to define a max limitation of M_{A}. If M_{A} is incremented to a large number, the decaying factor

ACK/NACK	A	A	A	N	A	A	N	N	A	A	A	A
L_A	1	2	3	0	1	2	0	0	1	2	3	4
M_A	1	2	3	0	1	2	0	0	1	2	3	3
M_N	0	0	0	1	0	0	1	2	0	0	0	0

Fig. 4: An Example of momentum index M_{A} and M_{N}. M_{A} is increased by 1 if ACK is received and M_{N} is increased by 1 if NACK is received. M_{A} will not be increased when M_{A} will be greater than $M_{\mathrm{A,\,Max}}$ (assume $M_{\mathrm{A,\,Max}} = 3$). L_{A} denotes the length of ACK sequences and it has no maximum limitation.

$_{\mathrm{down}}^{M_{\mathrm{A}}[t]}$ in (7) approaches zero. As a result, the backoff factor $_{\mathrm{OLLA}}$ can only be adjusted in a limited range, ultimately impacting the performance of link adaptation. The update of M_{A} can be written as equation listed below:

$$M_{\mathrm{A}}[t] = \begin{cases} \min(M_{\mathrm{A}}[t \quad 1] + 1; M_{\mathrm{A,\,Max}}) & \text{if ACK} \\ 0 & \text{if NACK} \end{cases} \quad (8)$$

Similarly, M_{N} can be updated as following:

$$M_{\mathrm{N}}[t] = \begin{cases} \min(M_{\mathrm{N}}[t \quad 1] + 1; M_{\mathrm{N;Max}}) & \text{if NACK} \\ 0 & \text{if ACK} \end{cases} \quad (9)$$

Algorithm 2 outlines the working procedure of NOLLA.

Algorithm 2 Non-Linear Outer Loop Link Adaptation

Initialize: Define $_{\mathrm{init}}$, $\mathrm{BLER}_{\mathrm{target}}$, $_{\mathrm{up}}$ and $_{\mathrm{down}}$.
$_{\mathrm{OLLA}} = 0$, $_{\mathrm{down}} = {}_{\mathrm{init}}$, $_{\mathrm{up}} = \frac{1-\mathrm{BLER}_{\mathrm{target}}}{\mathrm{BLER}_{\mathrm{target}}}\ {}_{\mathrm{init}}$
for Time slots $t = 1$ to N_{TTI} **do**
  if $t \mod T_{\mathrm{CQI}} = 0$ **then** . T_{CQI} denotes CQI interval
    Update estimated SINR $\hat{}$ based on new CQI.
  end if
  if ACK received **then**
    Update: M_{A} according to (8) and set $M_{\mathrm{N}} = 0$
    Update: $_{\mathrm{OLLA}}[t] = {}_{\mathrm{OLLA}}[t \quad 1] \quad {}_{\mathrm{down}}^{M_{\mathrm{A}}[t]} \quad {}_{\mathrm{down}}$
  else if NACK received **then**
    Update: M_{N} according to (9) and set $M_{\mathrm{A}} = 0$
    Update: $_{\mathrm{OLLA}}[t] = {}_{\mathrm{OLLA}}[t \quad 1] + {}_{\mathrm{up}}^{M_{\mathrm{N}}[t]} \quad {}_{\mathrm{up}}$
  end if
  calculate effective SINR: $\hat{}_{\mathrm{eff}} = \hat{} \quad {}_{\mathrm{OLLA}}[t]$
  select MCS according to $\hat{}_{\mathrm{eff}}$ based on Eq. (1).
end for

C. Parameter Choice for NOLLA

In the NOLLA algorithm, four parameters, M_A, M_N, $_{\mathrm{up}}$ and $_{\mathrm{down}}$ are introduced. M_A and M_N are random variables that depend on the HARQ feedback sequence, and the exponential bases $_{\mathrm{up}}$ and $_{\mathrm{down}}$ would be predefined. This section will explain the criteria for choosing these parameters.

For OLLA, Eq. (6) describes the relationship between $_{\mathrm{up}}$ and $_{\mathrm{down}}$. Step sizes of NOLLA become ${}_{\mathrm{down}}^{M_A} \cdot {}_{\mathrm{down}}$ and ${}_{\mathrm{up}}^{M_N} \cdot {}_{\mathrm{up}}$. Therefore, the new parameters of NOLLA should fulfill the following equation:

$$\frac{\mathrm{E}\{ {}_{\mathrm{up}}^{M_{\mathrm{N}}} \} \cdot {}_{\mathrm{up}}}{\mathrm{E}\{ {}_{\mathrm{down}}^{M_{\mathrm{A}}} \} \cdot {}_{\mathrm{down}}} = \frac{1-\mathrm{BLER}_{\mathrm{target}}}{\mathrm{BLER}_{\mathrm{target}}} \quad (10)$$

According to the definition, M_{A} and M_{N} are integer valued random variable ranging from 0 to $M_{\text{A,MAX}}$ and $M_{\text{N,MAX}}$, respectively. Taking ${}^{M_{\text{A}}}_{\text{down}}$ as an example, the expectation can be calculated through the following equation:

$$\mathsf{E}\{\ {}^{M_{\text{A}}}_{\text{down}}\} = \sum_{i=0}^{M_{\text{A, Max}}} p(M_{\text{A}} = i) \cdot \ {}^{i}_{\text{down}} \tag{11}$$

where $p(M_{\text{A}} = i)$ is the probability of M_{A} being equal to i . In order to derive the probabilities of M_{A}, an intermediate variable L_{A} is introduced to denote the length of ACK sequences, as shown by the example in Fig. 4. The state transition diagrams are depicted in Fig. 5.

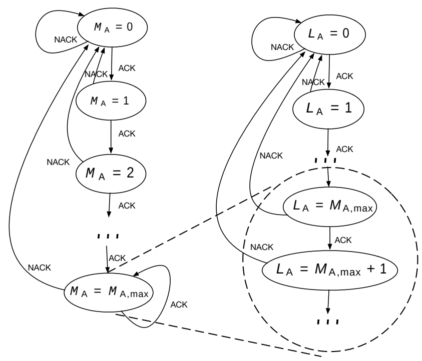

Fig. 5: The state transition diagrams of M_{A} and L_{A}.

With the error probability denoted by P_{e}, the probabilities of $M_{\text{A}} = i$ and $L_{\text{A}} = i$ for $i < M_{\text{A,max}}$ are given by:

$$p(M_{\text{A}} = i) = p(L_{\text{A}} = i) = P_{\text{e}}(1 - P_{\text{e}})^i \tag{12}$$

The summation of $p(L_{\text{A}} = i)$ can be calculated:

$$\sum_{i=0}^{\infty} p(L_{\text{A}} = i) = \lim_{n\to\infty} \sum_{i=0}^{n} P_e \cdot (1 - P_e)^i = 1 \tag{13}$$

The probability for $M_{\text{A}} = M_{\text{A,Max}}$ can be obtained:

$$\begin{aligned} p(M_{\text{A}} = M_{\text{A,Max}}) &= p(L_{\text{A}} \geqslant M_{\text{A,Max}}) = \sum_{i=M_{\text{A,Max}}}^{\infty} p(L_{\text{A}} = i) \\ &= \sum_{i=0}^{\infty} p(L_{\text{A}} = i) - \sum_{i=0}^{M_{\text{A,Max}}-1} p(L_{\text{A}} = i) \\ &= 1 - \sum_{i=0}^{M_{\text{A,Max}}-1} p(M_{\text{A}} = i) \\ &= (1 - P_{\text{e}})^{M_{\text{A,Max}}} \end{aligned} \tag{14}$$

The probability mass function (PMF) for M_{A} can be written as:

$$p(M_{\text{A}} = i) = \begin{cases} P_e \cdot (1 - P_e)^i & \text{if } i < M_{\text{A;Max}} \\ (1 - P_{\text{e}})^{M_{\text{A,Max}}} & \text{if } i = M_{\text{A;Max}} \end{cases} \tag{15}$$

Similarly the expected value of decaying factor for ${}_{\text{down}}$ is calculated as:

$$\mathsf{E}\{\ {}^{M_N}_{\text{up}}\} = \sum_{i=0}^{M_{\text{N, Max}}} p(M_{\text{N}} = i) \cdot \ {}^{i}_{\text{up}} \ ; \tag{16}$$

TABLE I: Simulation settings for link transmission and link adaptation

CQI quatiziation	4 bits uniform
CQI Interval	5 TTIs
HARQ	no delay and error free
No. subcarriers of transport block	72
Subcarrier spacing	15 kHz
No. Symbols of transport block	14
Symbols durations	66.67 μs
MCS	[11]
Δ_{init}	0.1
$\text{BLER}_{\text{target}}$	0.1

and the PMF of M_{N} is given by:

$$p(M_{\text{N}} = i) = \begin{cases} (1 \quad P_e) \quad P_e^i & \text{if } i < M_{\text{N;Max}} \\ P_e^{M_{\text{N;Max}}} & \text{if } i = M_{N;\text{Max}} \end{cases} \tag{17}$$

To wrap up, for NOLLA, newly introduced parameters need to follow the relationship given by Eq. (10). The contained expectation values are calculated using Eq. (11) and (16). The corresponding PMFs are derived in Eq. (15) and (17).

IV. Numerical Results

A. Simulation Configuration

Numerical simulation is conducted in an orthogonal frequency division multiplexing (OFDM) system to evaluate the reliability and efficiency of the proposed NOLLA algorithm and the conventional OLLA algorithm. As shown in table I, CQI feedback interval T_{CQI} is set to 5 TTIs and HARQ feedback is assumed to be sent without delay and errors. A single transport block contains 72 subcarriers and 14 OFDM symbols. The subcarrier spacing is set as 15 kHz and symbol duration as 66.7 s according to numerology 0 defined in [10].

In our work, two simulation scenarios are set up for evaluating LA algorithms. Firstly, channel is set as a additive white Gaussian noise (AWGN) channel with a static SINR of 10.5 dB, but realization for each link at each TTI is independent. Then, as depicted in Fig. 6, a scenario with N_{I} interference sources will be simulated. In this case, SINR can be written using the following equation:

$$= \frac{P_S}{P_N + \sum_{i=1}^{N_{\text{I}}} P_{\text{I};i}} \tag{18}$$

where P_S is the power of received signal, P_N is the noise power. $P_{\text{I};i}$ indicates the interference power from i-th interference source. N_{I} is assumed to be 2 in our case. Corresponding values and periodical patterns can be found in table II. Taking interference 1 as an example, it will transmit a signal for every 10 TTIs, and each time last for two TTIs.

B. Results and Analysis

1) Environment with AWGN channel: In this part, we consider an AWGN channel with static SINR set to 10.5 dB. At first, we observe the behavior of single link transmission

TABLE II: Parameter for interference model

	Interval	Duration	Normalized Power
Interference 1	10	2	0.02
Interference 2	15	3	0.05
Noise			0.1
Signal			1

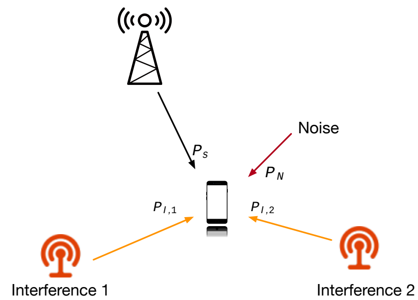

Fig. 6: Simulation scenarios with periodical interference pattern.

as presented in Fig. 7. Both OLLA and NOLLA have "climb and fall" phases when estimating effective SINR ^$_{\text{eff}}$. Since NOLLA adjusts the step sizes with decaying factors, the climb phase of is elongated and less NACKs are received overall. The fluctuation of ^$_{\text{eff}}$ occurs even without receiving NACKs, because ^ is updated based on the newly received CQI.

In Fig. 8, the cumulative distribution functions (CDF) of BLER of different algorithms are depicted. NOLLA-matched method ($_{\text{up}}$ = 0:77 and $_{\text{down}}$ = 0:92) means NOLLA algorithm with parameters which can meet the relationships derived in section III-C. In order to obtain matched parameters, we search a reasonable $_{\text{up}}$ at first, then corresponding $_{\text{down}}$ is calculated. For the NOLLA-unmatched ($_{\text{up}}$ = 0:9 and $_{\text{down}}$ = 0:9) means that the parameters of NOLLA are

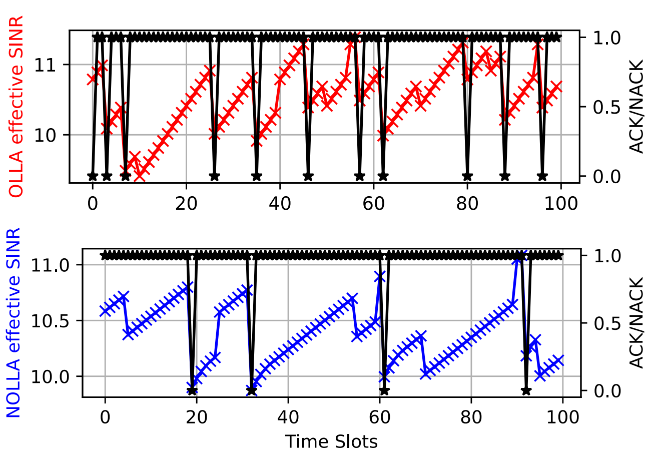

Fig. 7: Effective SINR estimate of OLLA (upper) and NOLLA (lower). Black curves are HARQ feedbacks. 1 means ACK and 0 means NACK. ^$_{\text{eff}}$ of NOLLA has longer climbing phase. NACKs are less frequent for NOLLA. ^$_{\text{eff}}$ fluctuates without NACKs due to updating ^ with newly received CQI.

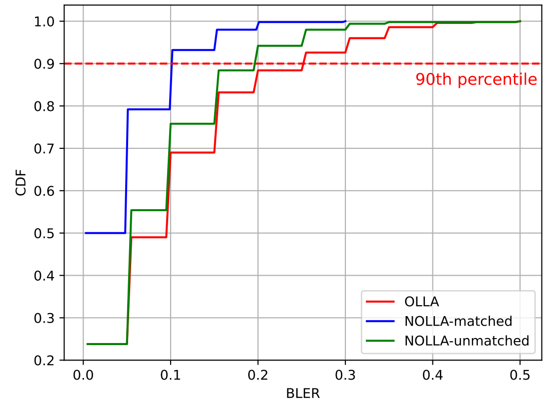

Fig. 8: CDF of BLER of link adaptation algorithms with static SINR. The further to the left the curve is, the lower the BLER for the corresponding links.

chosen arbitrarily. We observe that OLLA has the highest BLERs. The BLER of NOLLA-matched is the lowest. The performance of NOLLA-unmatched is in between. For instance, 90% of NOLLA-matched links have BLERs below 0.1 and the corresponding BLER for OLLA is 0.25. It is worth mentioning that curves in Fig. 8 have stepped shapes the number of simulation links is 20, therefore the granularity on x-axis is 0.05.

CDFs of throughput are presented in Fig. 9. It is obvious NOLLA-matched is able to achieve higher efficiency than OLLA and NOLLA-unmatched. More specifically, 90% of NOLLA-matched links can reach throughputs more than approximately 1.85 bps/Hz, while 90% of OLLA links are more than roughly 1.63 bps/Hz. One interesting phenomenon is that OLLA and NOLLA-unmatched have a stronger ability to reach a higher throughput, over 2.2 bps/Hz. The reason is that, OLLA and NOLLA-unmatched have less conservative policies when adjusting SINR and choosing MCS, therefore, in some cases, higher MCS will be selected. If transmissions with higher MCS are received successfully, higher efficiency will be achieved, but the cost is a higher BLER which has been shown in Fig. 8.

2) Environment with Interference: In Fig. 10, CDFs of BLERs for NOLLA (matched) and OLLA are shown. Compared with the performance under AWGN channel in Fig. 8, BLERs of OLLA of NOLLA are both increased because SINR is fluctuated with interference and it is more challenging for LA algorithms to select proper MCS. However, NOLLA still keeps the advantages. 90% of NOLLA links have BLERs under 0.15 and 90% of OLLA links have BLERs below 0.3.

CDFs of throughputs are drawn in Fig. 11. In the high throughputs area (throughputs greater than 1.7 bps/Hz), curves of OLLA and NOLLA are overlapped. NOLLA is helpful for preventing errors caused by overestimation, but errors in the scenario with interference happen because SINR fluctuates. Therefore, the advantages of NOLLA in the scenario with interference are not so significant as that in the scenario

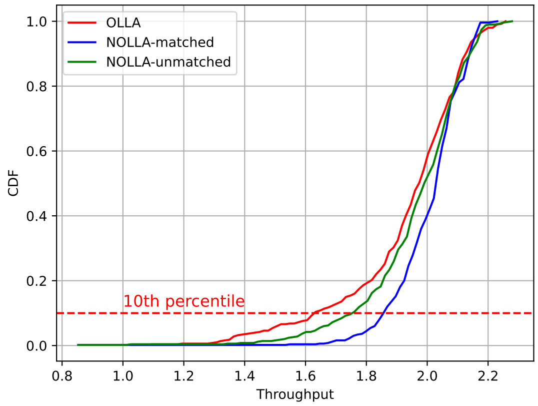

Fig. 9: CDF of throughput of link adaptation algorithms with static SINR. The further to the right the curve is, the higher throughput for the corresponding links.

with AWGN channel. Nevertheless, NOLLA is still better since most of NOLLA links have higher throughputs. 90% of NOLLA links are able to have throughputs greater than 1.55 bps/Hz. For OLLA, throughputs of 90% links are higher than 1.3 bps/Hz.

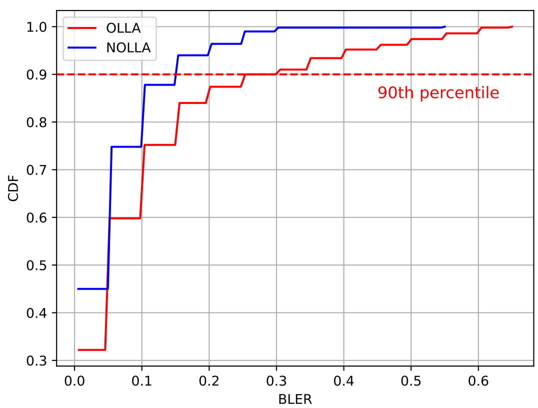

Fig. 10: CDF of BLER of link adaptation algorithms with periodical interference.The further to the left the curve is, the lower the BLER for the corresponding links.

V. Conclusion

In this paper, we investigated the enhancement of OLLA to address the issue of fluctuating signal-SINR estimates from OLLA that can negatively impact link transmission performance. In order to solve it, we propose the improved LA algorithm, NOLLA, which incorporates exponential decaying factors to adjust the estimates on SINR in a more flexible way. Therefore NOLLA could minimize the number the error caused by overestimation. We also evaluated the performance numerically in the scenarios with static SINR and periodic interference. The results show that NOLLA is able to reach higher throughput and keeping lower error rates at the same time. Finally, the proposed algorithm is highly compatible

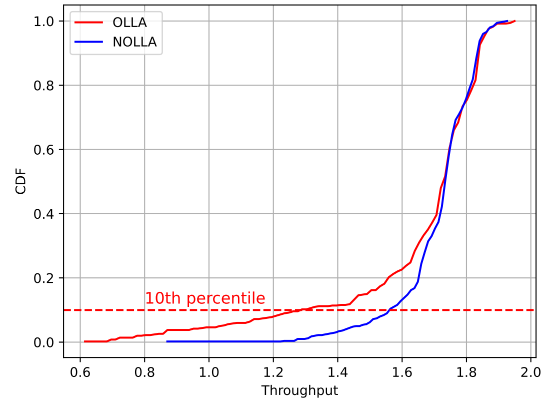

Fig. 11: CDF of throughput of link adaptation algorithms with periodical interference. The further to the right the curve is, the higher throughput for the corresponding links.

with OLLA and has the potential to significantly improve the performance of wireless communication systems that require high reliability and throughput with minimal additional effort.

References

[1] Saqib Hakak, Thippa Reddy Gadekallu, Praveen Kumar Reddy Maddikunta, Swarna Priya Ramu, M Parimala, Chamitha De Alwis, and Madhusanka Liyanage. Autonomous vehicles in 5g and beyond: A survey. *Vehicular Communications*, page 100551, 2022.

[2] Vitaly Petrov, Margarita Gapeyenko, Stefano Paris, Andrea Marcano, and Klaus I Pedersen. Extended reality (xr) over 5g and 5g-advanced new radio: Standardization, applications, and trends. *arXiv preprint arXiv:2203.02242*, 2022.

[3] Gwanmo Ku and John MacLaren Walsh. Resource allocation and link adaptation in lte and lte advanced: A tutorial. *IEEE communications surveys & tutorials*, 17(3):1605–1633, 2014.

[4] Klaus I Pedersen, Guillaume Monghal, Istvan Z Kovacs, Troels E Kolding, Akhilesh Pokhariyal, Frank Frederiksen, and Preben Mogensen. Frequency domain scheduling for ofdma with limited and noisy channel feedback. In *2007 IEEE 66th Vehicular Technology Conference*, pages 1792–1796. IEEE, 2007.

[5] Francisco Blanquez-Casado, Maria del Carmen Aguayo Torres, and Gerardo Gomez. Link adaptation mechanisms based on logistic regression modeling. *IEEE Communications Letters*, 23(5):942–945, 2019.

[6] Sachin Nayak and Sumit Roy. Novel markov chain based urllc link adaptation method for 5g vehicular networking. *IEEE Transactions on Vehicular Technology*, 70(12):12302–12311, 2021.

[7] Zhijie Dong, Junchao Shi, Wenjin Wang, and Xiqi Gao. Machine learning based link adaptation method for mimo system. In *2018 IEEE 29th Annual International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC)*, pages 1226–1231. IEEE, 2018.

[8] Raja Karmakar, Samiran Chattopadhyay, and Sandip Chakraborty. Smartla: Reinforcement learning-based link adaptation for high throughput wireless access networks. *Computer Communications*, 110:1–25, 2017.

[9] Vidit Saxena and Joakim Jaldén. Bayesian link adaptation under a bler target. In *2020 IEEE 21st International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pages 1–5. IEEE, 2020.

[10] 3GPP. Physical channels and modulation. Technical Specification (TS) 38.211, 3rd Generation Partnership Project (3GPP), 7 2020. Version 16.2.0.

[11] 3GPP. Physical layer procedures for data. Technical Specification (TS) 38.214, 3rd Generation Partnership Project (3GPP), 10 2018. Version 15.3.0.