This is the author's version of an article that has been published in this journal.

Changes were made to this version by the publisher prior to publication.

The final version of record is available at https://doi.org/10.1109/INDIN51400.2023.10218083

Linear Combination of Exponential Moving Averages for Wireless Channel Prediction

Gabriele Formis^{*†}, Stefano Scanzio^{*}, Gianluca Cena^{*}, and Adriano Valenzano^{*}, National Research Council of Italy (CNR–IEIIT), Italy. Email: gabriele.formis@ieiit.cnr.it, stefano.scanzio@cnr.it, gianluca.cena@cnr.it, adriano.valenzano@cnr.it

[†]Politecnico di Torino, Italy

Abstract—The ability to predict the behavior of a wireless channel in terms of the frame delivery ratio is quite valuable, and permits, e.g., to optimize the operating parameters of a wireless network at runtime, or to proactively react to the degradation of the channel quality, in order to meet the stringent requirements about dependability and end-to-end latency that typically characterize industrial applications.

In this work, prediction models based on the exponential moving average (EMA) are investigated in depth, which are proven to outperform other simple statistical methods and whose performance is nearly as good as artificial neural networks, but with dramatically lower computational requirements. Regarding the innovation and motivation of this work, a new model that we called EMA linear combination (ELC), is introduced, explained, and evaluated experimentally. Its prediction accuracy, tested on some databases acquired from a real setup based on Wi-Fi devices, showed that ELC brings tangible improvements over EMA in any experimental conditions, the only drawback being a slight increase in computational complexity.

I. INTRODUCTION

New paradigms like Industry 4.0 [1] and beyond [2] demand for industrial networks with high flexibility and the ability to suitably tackle heterogeneity [3]. In this respect, wireless communication technologies are one of the main enablers, by supporting mobility and automatic (re)configurability of industrial systems [4]. The ever increasing need for wire-free solutions in industries, homes, and cities, implies that specific research activities, aimed at making the related technologies capable to fulfill the very strict requirements, in terms of dependability and end-to-end latency, of industrial networks [5], [6], [7], [8], [9], [10], are more and more demanded. Besides communication, services like localization [11], timesynchronization [12], and roaming [13] have to be also provided for enabling fully wireless industrial systems (required, e.g., when devices are characterized by mobility), by adapting the existing approaches to the specific wireless technology.

If the future quality of a wireless link could be predicted in advance with a certain accuracy, e.g., in terms of frame delivery ratio (FDR), the communication protocol could exploit this information for trying to meet the specific requirements of industrial applications. For instance, when the quality of a specific channel is expected to worsen, the application can switch to another channel (suffering from less disturbance) to preserve communication reliability [14] or to prevent energy consumption from increasing [15]. As an alternative, it can change some network/communication parameters or diminish the amount of best-effort traffic (shaping) to privilege timesensitive data exchanges. Among the many wireless technologies available today, Wi-Fi and 5G are the most promising for applications with higher demands in terms of dependability and end-to-end latency, while allowing at the same time high throughput. This work is based on Wi-Fi, but the technique we present for channel quality prediction can be applied with minimal changes to other wireless communication technologies as well.

Many research works made use of machine learning (ML) for channel quality prediction in Wi-Fi [16]: in [17] artificial neural networks (ANN) were applied to artificial data; in [18], [19] they were exploited on data derived from real devices; in [20], [21], [22] ML is used for the prediction of the channel gain and/or the received signal strength. The use of ANNs is quite expensive from the computational point of view, and consequently in [23] the benefits provided by less CPU-hungry approaches like moving averages and regression models were analyzed in depth. That work showed that the exponential moving average (EMA) is able to offer the best performance among non-ANN models, and behaved almost as well as an ANN trained only on frame losses. Quite interestingly, EMA featured the lowest computational complexity among the analyzed models. This makes it suitable for implementation in any kind of (embedded) device.

In this paper, an algorithm named EMA linear combination (ELC) is presented and evaluated over a 50-day-long test database, acquired in a real setup that includes several Wi-Fi nodes operating on different channels. The optimal parameters of this algorithm are found by minimizing the mean squared error for the training database. Results show that ELC provides consistent improvements when compared to the simpler EMA model, yet keeping computational complexity relatively low. The following Section II introduces the concept of channel prediction and the EMA model analytically, and Section III describes the new ELC algorithm, which is evaluated in Section IV, before the conclusive remarks of Section V.

This work was partially supported by the European Union under the Italian National Recovery and Resilience Plan (NRRP) of NextGenerationEU, partnership on "Telecommunications of the Future" (PE00000001 - program "RESTART").

II. CHANNEL PREDICTION AND EMA MODEL

This work considers a real Wi-Fi link between a STA and an AP. The wireless channel was probed periodically by means of confirmed one-shot transmissions (no retries allowed) with a period $T_s = 0.5$ s, corresponding to a frequency of 2 Hz. The receiver node (the AP, in this case) confirms the correct arrival of the *i*-th frame by returning an Acknowledgement (ACK) frame. The correct reception of this ACK frame by the sender node is recorded as the outcome $x_i = 1$ (success). Otherwise, if the ACK frame is not received, we logged $x_i = 0$ (failure). The above measurement procedure was practically implemented in real Wi-Fi devices by following the indications reported in Subsection IV-A.

A database consists of an ordered sequence of outcomes $\mathcal{D} = \{x_1, ..., x_i, ..., x_{|\mathcal{D}|}\}$, obtained from the experimental testbed on a time interval long enough (some days). Two kinds of databases were used in this work: *training* (\mathcal{D}_{tr}) , used to train the models by learning some parameters from data, and *test* (\mathcal{D}_{te}) , used to evaluate the proposed models. Their sizes, in terms of the number of outcomes, are $|\mathcal{D}_{tr}|$ and $|\mathcal{D}_{te}|$, respectively.

The goal of this work is finding a simple and effective way to predict the quality of the wireless channel in terms of the FDR, i.e., the ratio between the number of ACK frames that come back to the sender node and the number of transmitted frames. We relied on the exponentially weighted moving average (EMA) of outcomes x_i , which is the simplest form of an infinite impulse response (IIR) low-pass filter, described as

$$y_i = \alpha \cdot x_i + (1 - \alpha) \cdot y_{i-1}, \tag{1}$$

where y_i is the current FDR prediction, y_{i-1} is the previous prediction, x_i is the current outcome, and α is a weight for balancing between present and past. An α value close to 1 privileges the current outcome (i.e., the model is more reactive to track sudden changes in channel quality), while values close to 0 prioritize the previous prediction made by EMA (making the output more stable and less susceptible to statistical fluctuations). For the very first prediction we selected the value $y_0 = 0.5$, but if a better estimation of the channel FDR is available its value should be used to initialize y_0 . In the Z-domain, the low-pass filter in (1) is characterized by the transfer function

$$H^{(\alpha)}(z) = \frac{Y(z)}{X(z)} = \frac{\alpha}{1 - (1 - \alpha)z^{-1}}.$$
 (2)

As highlighted in Fig. 1, in order to assess the performance of the proposed EMA model a *target* is computed for every value of x_i in \mathcal{D}_{te} , with $1 \le i \le |\mathcal{D}_{te}| - N_f$, starting from the following N_f outcomes $(x_{i+1}, ..., x_{i+N_f})$

$$t_i = \frac{1}{N_f} \sum_{j=i+1}^{i+N_f} x_j.$$
 (3)

This target represents an estimate of the FDR over an interval in the immediate future whose width is $T_f = N_f \cdot T_s$. For



Fig. 1. Example of computation of the EMA model (y_i) , target (t_i) , and mean squared error (e_i^2) with i = 5 (the first N_s outcomes are used only for the execution of EMA model, but not for the computation of the error).

example, if $N_f = 3600$ then $T_f = 30 \text{ min}$, and the middle point of such interval is located 15 min after the current sample. It is not worth shrinking N_f too much because of statistical fluctuations, which would make the target unreliable.

The most recent value y_i given by (1) was taken as the prediction for the target, under the reasonable assumption that, despite they are displaced by $T_f/2$, it constitutes the best possible estimate. The adoption of predictive models like those based on linear and polynomial regression has been analyzed in [23], where their showed poorer prediction accuracy than EMA.

Evaluation of the prediction accuracy was carried out on outcomes with index $i = N_s + 1, N_s + 2, ..., |\mathcal{D}_{te}|$ of \mathcal{D}_{te} . The first N_s outcomes were discarded because we aim to evaluate steady-state accuracy, neglecting the initial (and potentially long) transient from y_0 up to when FDR estimation settles. The closer y_i and t_i are, the better the prediction ability of the model. To quantify the model's accuracy, several kinds of errors were considered: the real error $e_i = t_i - y_i$, the absolute error $|e_i| = |t_i - y_i|$, and the squared error $e^2 = (t_i - y_i)^2$. The latter is the one that is minimized by the learning method described in Section III.

Starting from these errors, statistical indices can be computed, which include the average (μ_e , $\mu_{|e|}$ and μ_{e^2}), standard deviation (σ_e , $\sigma_{|e|}$ and σ_{e^2}), minimum (e_{\min} , $|e|_{\min}$ and e_{\min}^2), percentiles (e_{p90} , e_{p95} , e_{p99} , $|e|_{p90}$, $|e|_{p95}$, $|e|_{p99}$ and e_{p90}^2 , e_{p95}^2 , e_{p99}^2), and maximum (e_{\max} , $|e|_{\max}$ and e_{\max}^2). The number of predictions on which these statistics are computed is $|\mathcal{D}_{te}| - N_s - N_f$. As an example, the mean squared error (MSE) can be obtained as $\mu_{e^2} = \sum_{i=N_s+1}^{|\mathcal{D}|-N_f} (t_i - y_i)^2$.

The EMA model is parameterized by a single quantity, the α weight. Optimal values can be found for it, each of which minimizes some objective functions for some database. We denote α^* the value that minimizes the MSE (μ_{e^2}) for \mathcal{D}_{tr}

$$\alpha^* = \arg\min_{\alpha} \sum_{i=N_s+1}^{|\mathcal{D}_{tr}|-N_f} \left(t_i - y_i\right)^2.$$
(4)

To assess in a fair way EMA performance, the value α^* estimated this way is then used on the test database \mathcal{D}_{te} .

It is worth pointing out that optimality of α^* refers to the whole \mathcal{D}_{tr} ($|\mathcal{D}_{tr}| - N_s - N_f$ values). However, larger values for the α weight permit to track FDR variations more quickly, while smaller values provide better accuracy when channel

conditions are stationary. In addition, α^* was estimated on \mathcal{D}_{tr} , but it cannot be assured to be optimal for \mathcal{D}_{te} as well.

A mechanism for finding the different α values that represent at best the different FDR variation patterns for a given database \mathcal{D} will be presented in Subsection III.

III. LINEAR COMBINATION OF EMA

In this section, the EMA linear combination (ELC) algorithm is presented and analyzed. This algorithm is composed of two main parts: α and λ selection (λ -s) based on \mathcal{D}_{tr} , and evaluation based on \mathcal{D}_{te} . Steps are summarized in the example of Fig. 2, which is meant to clarify ELC operation.

A. λ -s: Initial selection of α weights

In the first step of the algorithm, a number of EMA models were executed concurrently with different values of the α weight, taken from a *starting sequence* α S (i.e., an ordered sequence of numbers) defined as

$$\alpha \mathbf{S} = (\alpha_n \mid \alpha_n = \alpha^* \cdot r^n), \tag{5}$$

where $\alpha_n = \alpha^* \cdot r^n$ is sort of a finite-size geometric progression with common ratio r in which n can also assume negative values. In particular, $n \in [-N_l, -N_l +$ $1, ..., -1, 0, 1, ..., N_u - 1, N_u]$, where N_l and N_u are used to specify the lower and upper bound for the selection of α weights, respectively. The length of the sequence αS is $|\alpha S| = N_l + 1 + N_u$. As an example, when r = 2, $N_l = 2$, and $N_u = 4$, the starting sequence includes 7 values and is $\alpha S = (\frac{\alpha^*}{4}, \frac{\alpha^*}{2}, \alpha^*, 2\alpha^*, 4\alpha^*, 8\alpha^*, 16\alpha^*)$.

Since the purpose of the λ -s algorithm is to select the set of α weights that are more suitable for a specific database, the space of α weights should be sampled in such a way not to discard any relevant values. Using a geometrical progression permits to have more α weights near α^* , progressively making them less frequent when the distance from α^* increases. This selection strategy was verified experimentally and compared to a uniform sampling of the α space. Results, not reported here for space reasons, confirm that this choice was the most appropriate.

Decreasing r permits to analyze the space of α weights with finer granularity, but execution time grows because the number of EMA models to be checked is higher (given the lower and upper bounds). N_l and N_u were selected large enough to embrace values of α that are three orders of magnitude smaller and larger than α^* , respectively. Let $(y_i^{(\alpha_1)}, y_i^{(\alpha_2)}, ..., y_i^{(\alpha_{|\alpha|})})$ be the sequence of predictions

Let $(y_i^{(\alpha_1)}, y_i^{(\alpha_2)}, ..., y_i^{(\alpha_{\lfloor \alpha S \rfloor})})$ be the sequence of predictions obtained with the EMA models parameterized according to the values in sequence αS . For the example above, $\alpha_1 = \frac{\alpha^*}{4}, \alpha_2 = \frac{\alpha^*}{2}, \alpha_3 = \alpha^*, ..., \alpha_{|\alpha S|} = 16\alpha^*$.

B. λ -s: Linear combination

To improve prediction accuracy, a linear combination (COM) can be evaluated from the output values $y_i^{(\alpha_j)}$ produced by several EMA models, each one characterized by weight α_j ,

$$y_i^{\text{COM}} = \sum_{j=1}^{|\alpha A|} \lambda^{(\alpha_j)} \cdot y_i^{(\alpha_j)}, \qquad (6)$$

where coefficients $\lambda^{(\alpha_j)}$ are selected so that

$$\sum_{j=1}^{|\alpha A|} \lambda^{(\alpha_j)} = 1, \tag{7}$$

and $\alpha A = (\alpha_1, \alpha_2, ..., \alpha_{|\alpha A|})$ represents in a generic way the sequence of α weights of the considered EMA models. Index j in (6) selects both the specific EMA model (characterized by the value of weight α_j , which determines the associated prediction $y_i^{(\alpha_j)}$) and the coefficient $\lambda^{(\alpha_j)}$ used to weight its contribution in the linear combination.

The COM model in (6) coincides with a multipole low-pass IIR filter characterized by the transfer function

$$H^{(\alpha \mathbf{A})}(z) = \sum_{j=1}^{|\alpha \mathbf{A}|} \lambda^{(\alpha_j)} \cdot H^{(\alpha_i)}(z).$$
(8)

As analysed in a very preliminary way in [23], the linear combination of EMA models may offer better prediction accuracy than any one of them considered separately. In that paper, the sequence of α weights was statically selected as $\alpha A = (\frac{\alpha^*}{3}, \alpha^*, 3\alpha^*)$ and the three models were equally weighted $(\lambda^{(\alpha_1)} = \lambda^{(\alpha_2)} = \lambda^{(\alpha_3)} = \frac{1}{3})$. In this paper that solution is tangibly enhanced thanks to the ELC algorithm, which permits the automatic selection of both the α_j weights and $\lambda^{(\alpha_j)}$ coefficients of the linear combination by means of a suitable training phase performed on a relevant database.

C. λ -s: Minimizing the MSE

Among the potentially many α values included in the starting sequence α S, we aim to assess which ones affect prediction accuracy more. The selection of a small subset of α S, which includes the most relevant EMA weights, has been addressed as a minimization problem of the mean squared error μ_{e^2} of the COM model (6), constrained by (7) and with the boundaries

$$0 \le \lambda^{(\alpha_j)} \le 1,\tag{9}$$

over the training database \mathcal{D}_{tr} .

The method chosen to perform the minimization was the L-BFGS-B [24], a quasi-Newton optimization algorithm. It is a gradient-based optimization algorithm that iteratively updates an estimate of the solution using gradient and curvature information, while also handling bound constraints. It is designed for high-dimensional problems and can efficiently find the minimum of a function.

The outcome of the MSE minimization problem is a sequence $\lambda S = (\lambda^{(\alpha_1)}, \lambda^{(\alpha_2)}, ..., \lambda^{(\alpha_{|\alpha S|})})$ of optimal coefficients for the linear combination, each one associated to a specific value in the sequence αS (by definition, the size of sequences αS and λS is the same, i.e., $|\alpha S| = |\lambda S|$).

D. λ -s: Final selection of α weights

We observed that part of the combination coefficients evaluated by the minimization algorithm in the previous Subsection III-C were rather small. Consequently, their contribution to the final prediction is irrelevant. This is even more true if



Fig. 2. Example about the application of the ELC algorithm ("min" refers to the minimization of the linear combination, as explained in Subsection III-C).

they are considered on a database different from the one used to perform minimization, e.g., \mathcal{D}_{te} . In addition, if the αS and λS sequences are shortened, the algorithm becomes faster and more suitable to be implemented in embedded devices with scarce computational power.

Starting from αS , a new *final sequence* αF is obtained by selecting part of the values α_i in αS for which the corresponding $\lambda^{(\alpha_i)}$ combination coefficients are larger. Practically, let $\overline{\alpha S} = (\overline{\alpha}_1, \overline{\alpha}_2, ..., \overline{\alpha}_{|\alpha S|})$ be a version of the sequence αS sorted in decreasing order of the associated value of $\lambda^{(\overline{\alpha}_i)}$. Let $N_{\alpha} = |\alpha F|$ represent the number of EMA models that are eventually selected for testing. It can be evaluated as

$$N_{\alpha} = \min\left(i \in 1... |\overline{\alpha S}| \mid \sum_{j=1}^{i} \lambda^{(\overline{\alpha}_{j})} \ge \lambda_{\max}\right), \qquad (10)$$

where λ_{max} is a threshold used to discriminate which weights must be included in αF . Sequence αF can then be derived as

$$\alpha \mathbf{F} = \left(\overline{\alpha}_i \in \overline{\alpha \mathbf{S}} \mid i \le N_\alpha\right). \tag{11}$$

For instance, setting $\lambda_{\text{max}} = 0.9$ means to take the minimum number of elements from αS so that the sum of the related coefficients in λS exceeds 90%. This is the same as keeping only the EMA filters that mostly contributed to reducing the MSE (μ_{e^2}).

The minimization algorithm is then applied a second time to the sequence αF , to obtain the new $\lambda^{(\alpha_i)}$ coefficients that minimize the MSE for \mathcal{D}_{tr} . These new $\lambda^{(\alpha_j)}$ values constitute the actual final sequence λF , which will be used for testing. Setting $\lambda_{\max} = 1.0$ implies that the second minimization of the linear combination is not performed.

E. Evaluation

Given the new final sequence $\alpha F = (\alpha_1, \alpha_2, ..., \alpha_{|\alpha F|})$ and $\lambda F = (\lambda^{(\alpha_1)}, \lambda^{(\alpha_2)}, ..., \lambda^{(\alpha_{|\lambda F|})})$, the linear combination in (6) provides the prediction y_i^{COM} of the COM model. The primary outcome of this work is to determine if the α weights and the $\lambda^{(\alpha_j)}$ coefficients of the linear combination, which are optimal for \mathcal{D}_{tr} by considering μ_{e^2} as the error metric, are also optimal for \mathcal{D}_{te} databases, by considering μ_{e^2} and, possibly, other error functions like μ_e .

IV. RESULTS

A. Database acquisition

The acquisition of \mathcal{D}_{tr} and \mathcal{D}_{te} was performed with the same experimental setup used in [23], which consists of four Wi-Fi adapters of type TP-Link TL-WDN4800 that are configured to transmit periodically (with a frequency of 2 Hz) frames of dimension 50 B directed to four different access points. The arrival of the ACK frame associated with every data frame is used to determine its transmission outcome (success or failure), which is then stored in a database.

Coherently with the rules of the software-defined MAC (SDMAC) framework [25], [26], the device driver was modified in order to: disable automatic retransmissions, disable frame aggregation, disable the backoff procedure, and fix the transmission speed to $54 \,\mathrm{Mb/s}$. These changes permitted us to sample the channel at a fixed rate, preventing the effects of some mechanisms in Wi-Fi that could distort measurements, for example by automatically retransmitting frames when the ACK does not arrive back to the sender node.

Training and test databases were acquired on four different channels, and their size is considerably larger than prior works on this topic like [23]. Acquisition spanned over 21.2 days for \mathcal{D}_{tr} and 12.8 days for \mathcal{D}_{te} on channel 1 (2.412 GHz), 21.9 days for \mathcal{D}_{tr} and 10.6 days for \mathcal{D}_{te} on channel 5 (2.432 GHz), 24.9 days for \mathcal{D}_{tr} and 15.2 days for \mathcal{D}_{te} on channel 9 (2.452 GHz), and 25.5 days for \mathcal{D}_{tr} and 15.2 days for \mathcal{D}_{te} on channel 13 (2.472 GHz).

B. EMA baseline

Plot 3.a reports the evolution of FDR for D_{te} referred to channel 13, computed with (3) and $N_f = 3600$. As can be seen, the quality of this Wi-Fi channel experienced consistent variations (from 60% to 80%), and sometimes disturbance spikes appeared suddenly. Channel 5 is characterized by a similar pattern, while channel 1 and, especially, channel 9 are characterized by faster and larger FDR variations (from 40% to 90%).

As highlighted in Plot 3.b, equation (4) is exploited to find the optimal values of α weights (that is, parameters α^*) for the four considered channels, as reported in the third column

 TABLE I

 Comparison between EMA and ELC for the considered WI-FI channels based on statistics about prediction errors.

Channel	Model	Model	μ_{e^2}	$e_{p_{95}}^2$	e_{\max}^2	$\mu_{ e }$	$\sigma_{ e }$	$\left e\right _{\mathrm{P90}}$	$ e _{P95}$	$\left e\right _{\mathrm{p}_{99}}$	$\left e\right _{\max}$	e_{\min}	e_{P5}	$e_{\rm P95}$	e_{\max}
		parameters	$[\cdot 10^{-3}]$				[%]					[%]			
1	EMA	$\alpha^* = 0.001$	2.62	13.88	101.51	3.38	3.85	8.64	11.78	16.93	31.86	-24.75	-8.16	9.14	31.86
	ELC	$N_{\alpha} = 5$	2.12	11.09	73.49	3.13	3.38	7.80	10.53	15.04	27.11	-22.97	-7.81	7.78	27.11
5	EMA	$\alpha^* = 0.000075$	1.07	2.71	74.61	2.17	2.44	4.31	5.20	10.73	27.31	-27.31	-4.56	4.12	7.58
	ELC	$N_{\alpha} = 3$	0.95	2.39	75.49	1.98	2.37	3.84	4.89	11.06	27.48	-27.48	-4.13	3.59	16.14
9	EMA	$\alpha^* = 0.000075$	8.62	42.60	363.64	5.77	7.27	13.61	20.64	34.91	60.30	-60.30	-20.51	11.21	21.01
	ELC	$N_{\alpha} = 2$	6.95	36.60	376.06	5.08	6.61	11.94	19.13	33.28	61.32	-61.32	-17.91	9.62	33.32
13	EMA	$\alpha^*=0.000625$	0.72	3.19	19.75	1.94	1.85	4.31	5.65	8.81	14.05	-12.07	-4.28	4.35	14.05
	ELC	$N_{\alpha} = 4$	0.69	2.88	22.60	1.95	1.77	4.16	5.37	8.53	15.03	-11.76	-4.00	4.35	15.03



Fig. 3. Example case of channel 13: timing diagram for FDR in \mathcal{D}_{te} (Plot 3.a), and computation of α^* from the MSE in \mathcal{D}_{tr} (Plot 3.b).

of Table I for rows labeled "EMA". Baseline results, obtained with the plain EMA model using these α^* values, are also reported in the table.

As expected, channels 13 and 5 show lower prediction errors, because they are characterized by smoother FDR variations. In these cases, the FDR can be predicted with an average absolute error $\mu_{|e|}$ of about 2%, and also high-order percentiles like $|e|_{p99}$ are bounded to prediction errors smaller than $\sim 10\%$. On the contrary, due to abrupt and difficult-to-predict FDR variations, the performance of channel 9 is rather poor, i.e., $\mu_{|e|} = 5.77\%$ and $|e|_{p99} = 34.91\%$.

Above considerations also hold for the other indicators about prediction accuracy, which are related to the real (e) and absolute (|e|) errors.

C. ELC

The parameters used as the initial selection of α weights for the evaluation of the ELC model (Subsection III-A) are $N_l = 17$, $N_u = 17$, and r = 1.5. Consequently, the starting sequence α S included $|\alpha$ S| = 35 α weights.

After the minimization step described in Subsection III-C, the sorted sequence of λ (i.e., $\overline{\lambda S}$) was used, in conjunction



Fig. 4. MSE (μ_{e^2}) vs. λ_{max} for the considered Wi-Fi channels (the value $\lambda_{max} = 75\%$ was selected).

with (10) and (11), to select the most significant coefficients $\lambda^{(\overline{\alpha}_j)}$ by varying the λ_{\max} parameter.

The plots in Fig. 4 report the mean squared error μ_{e^2} obtained with different values of λ_{\max} for the four training databases \mathcal{D}_{tr} related to channels 1, 5, 9, and 13. As can be seen, when $\lambda_{\max} > 75\%$ there are practically no improvements regarding μ_{e^2} . For this reason, the value $\lambda_{\max} = 75\%$ has been selected for the entire experimental campaign.

Rows labeled "ELC" in Table I show the results obtained for the four test databases \mathcal{D}_{te} that refer to channels 1, 5, 9, and 13, after all the steps of the ELC algorithm have been performed as described in Section III. The improvements this approach provides over plain EMA (a single optimized exponential moving average) are noticeable in all the four experimental conditions.

Results in Table I show that, excluding some of the maximum error values (e.g., e_{max}^2 and $|e|_{\text{max}}$ for channels 5, 9, and 13) and some high-order percentiles (e.g., $|e|_{p99}$ for channels 5 and 9), the ELC model practically always behaves better than the EMA model. In particular, the relative reduction of μ_{e^2} , which is the objective function minimized in this work, is quite consistent, and corresponds to 19.1%, 11.2%, 19.4%, and 4.2% for channels 1, 5, 9, and 13, respectively.

Regarding $\mu_{|e|}$, with the only exception of channel 13 (where the error remains quite stable), the relative reduction of the absolute error is 7.4%, 8.8%, and 12.0% for channels 1, 5, and 9, respectively. All these results confirm the higher prediction accuracy of the ELC-based model compared to EMA. Again, improvements concerning μ_{e^2} are better than those related to $\mu_{|\mu_e|}$, since the λ -s algorithm performs minimization on the MSE.

Finally, it is worth noting that the number N_{α} of EMA models that were selected by the λ -s algorithm slightly depends on the specific channel, but remains quite small when compared to the cardinality of the initial set of EMA models ($|\alpha S| = 35$). This means that ELC is not very demanding from the point of view of computation resources.

V. CONCLUSIONS

The ability to reliably predict the channel quality is the most important requirement for implementing wireless network protocols that proactively adapt their behavior to the characteristics of the communication medium. In this paper, a new method named ELC was presented and evaluated. By linearly combining EMA filters configured with different parameters, it permits to improve the accuracy with which the future quality of a wireless channel, expressed in terms of frame delivery ratio, is predicted.

The capabilities of the proposed method were experimentally assessed on a set of databases acquired from a real Wi-Fi installation, and a substantial reduction of the prediction error was observed with respect to the EMA model in every condition we analyzed. Moreover, the low computational complexity of ELC, compared to methods based on artificial neural network, makes it suitable for adoption in low-cost low-power devices where communication takes typically place on air, e.g., motes in wireless sensor networks (WSN).

Future research directions include the analysis of ELC in different experimental conditions, the use of ANNs for selecting frame-by-frame the best EMA model, and the use of ANNs for non-linear combinations of EMA models.

REFERENCES

- H. Cañas, J. Mula, M. Díaz-Madroñero, and F. Campuzano-Bolarín, "Implementing Industry 4.0 principles," *Computers and Industrial Engineering*, vol. 158, p. 107379, 2021.
- [2] P. K. R. Maddikunta, Q.-V. Pham, P. B, N. Deepa, K. Dev, T. R. Gadekallu, R. Ruby, and M. Liyanage, "Industry 5.0: A survey on enabling technologies and potential applications," *Journal of Industrial Information Integration*, vol. 26, p. 100257, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2452414X21000558
- [3] S. Scanzio, L. Wisniewski, and P. Gaj, "Heterogeneous and dependable networks in industry – A survey," *Computers in Industry*, vol. 125, p. 103388, 2021.
- [4] M. Friesen, L. Wisniewski, and J. Jasperneite, "Machine Learning for Zero-Touch Management in Heterogeneous Industrial Networks - A Review," in *IEEE 18th International Conference on Factory Communication Systems (WFCS 2022)*, 2022, pp. 1–8.
- [5] J. Perez-Ramirez, O. Seijo, and I. Val, "Time-Critical IoT Applications Enabled by Wi-Fi 6 and Beyond," *IEEE Internet of Things Magazine*, vol. 5, no. 3, pp. 44–49, 2022.
- [6] G. Cena, S. Scanzio, and A. Valenzano, "A Prototype Implementation of Wi-Fi Seamless Redundancy with Reactive Duplication Avoidance," in 2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA), vol. 1, 2018, pp. 179–186.

- [7] L. Leonardi, L. Lo Bello, and G. Patti, "LoRa support for long-range real-time inter-cluster communications over Bluetooth Low Energy industrial networks," *Computer Communications*, vol. 192, pp. 57–65, 2022.
- [8] G. Cena, S. Scanzio, and A. Valenzano, "Seamless Link-Level Redundancy to Improve Reliability of Industrial Wi-Fi Networks," *IEEE Trans. Ind. Informat.*, vol. 12, no. 2, pp. 608–620, April 2016.
- [9] J. Haxhibeqiri, P. A. Campos, I. Moerman, and J. Hoebeke, "Safetyrelated Applications over Wireless Time-Sensitive Networks," in *IEEE* 27th International Conference on Emerging Technologies and Factory Automation (ETFA 2022), 2022, pp. 1–8.
- [10] D. Cavalcanti, C. Cordeiro, M. Smith, and A. Regev, "WiFi TSN: Enabling Deterministic Wireless Connectivity over 802.11," *IEEE Communications Standards Magazine*, vol. 6, no. 4, pp. 22–29.
- [11] C. D. Fernandes, A. Depari, E. Sisinni, P. Ferrari, A. Flammini, S. Rinaldi, and M. Pasetti, "Hybrid indoor and outdoor localization for elderly care applications with lorawan," in 2020 IEEE International Symposium on Medical Measurements and Applications (MeMeA), 2020, pp. 1–6.
- [12] M. Mongelli and S. Scanzio, "A neural approach to synchronization in wireless networks with heterogeneous sources of noise," Ad Hoc Networks, vol. 49, pp. 1–16, 2016.
- [13] H. C. Yu, K. Alhazmi, and R. R. Rao, "Wi-Fi Roaming as a Locationbased Service," in *ICC 2020 - 2020 IEEE International Conference on Communications (ICC)*, 2020, pp. 1–7.
- [14] V. Kotsiou, G. Z. Papadopoulos, P. Chatzimisios, and F. Theoleyre, "LABeL: Link-Based Adaptive BLacklisting Technique for 6TiSCH Wireless Industrial Networks," in *Proceedings of the 20th ACM International Conference on Modelling, Analysis and Simulation of Wireless and Mobile Systems*, ser. MSWiM '17. Association for Computing Machinery, 2017, pp. 25–33.
- [15] G. Cena, S. Scanzio, and A. Valenzano, "Ultra-Low Power Wireless Sensor Networks Based on Time Slotted Channel Hopping with Probabilistic Blacklisting," *Electronics*, vol. 11, no. 3, 2022.
- [16] S. Szott, K. Kosek-Szott, P. Gawłowicz, J. T. Gómez, B. Bellalta, A. Zubow, and F. Dressler, "Wi-Fi Meets ML: A Survey on Improving IEEE 802.11 Performance With Machine Learning," *IEEE Communications Surveys and Tutorials*, vol. 24, no. 3, pp. 1843–1893, 2022.
- [17] A. K. Gizzini, M. Chafii, A. Nimr, and G. Fettweis, "Deep Learning Based Channel Estimation Schemes for IEEE 802.11p Standard," *IEEE Access*, vol. 8, pp. 113751–113765, 2020.
- [18] S. Scanzio, F. Xia, G. Cena, and A. Valenzano, "Predicting Wi-Fi link quality through artificial neural networks," *Internet Technology Letters*, vol. 5, no. 2, p. e326, 2022. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/itl2.326
- [19] S. Scanzio, G. Cena, C. Zunino, and A. Valenzano, "Machine Learning to Support Self-Configuration of Industrial Systems Interconnected over Wi-Fi," in *IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA 2022)*, 2022, pp. 1–8.
- [20] W. Jiang and H. D. Schotten, "Neural Network-Based Fading Channel Prediction: A Comprehensive Overview," *IEEE Access*, vol. 7, pp. 118112–118124, 2019.
- [21] W. Jiang, H. Dieter Schotten, and J.-y. Xiang, *Neural Network–Based Wireless Channel Prediction*. John Wiley & Sons, Ltd, 2020, ch. 16, pp. 303–325. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119562306.ch16
- [22] A. Kulkarni, A. Seetharam, A. Ramesh, and J. D. Herath, "DeepChannel: Wireless Channel Quality Prediction Using Deep Learning," *IEEE Trans. Veh. Technol.*, vol. 69, no. 1, pp. 443–456, 2020.
- [23] G. Formis, S. Scanzio, G. Cena, and A. Valenzano, "Predicting Wireless Channel Quality by means of Moving Averages and Regression Models," in 19th IEEE International Conference on Factory Communication Systems (WFCS 2023), 2023, pp. 1–8.
 [24] D. C. Liu and J. Nocedal, "On the limited memory BFGS
- [24] D. C. Liu and J. Nocedal, "On the limited memory BFGS method for large scale optimization," *Mathematical Programming*, vol. 45, p. 503–528, 1989. [Online]. Available: https://doi.org/10.1007/ BF01589116
- [25] G. Cena, S. Scanzio, and A. Valenzano, "SDMAC: A Software-Defined MAC for Wi-Fi to Ease Implementation of Soft Real-Time Applications," *IEEE Trans. Ind. Informat.*, vol. 15, no. 6, pp. 3143–3154, 2019.
- [26] —, "A software-defined MAC architecture for Wi-Fi operating in user space on conventional PCs," in 2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS), 2017, pp. 1–10.