Statistical BER Analysis of Wireline Links With Non-Binary Linear Block Codes Subject to DFE Error Propagation

Ming Yang, *Student Member, IEEE*, Shayan Shahramian, Hossein Shakiba, *Senior Member, IEEE*, Henry Wong, *Member, IEEE*, Peter Krotnev, and Anthony Chan Carusone, *Senior Member, IEEE*

***Abstract*— This paper presents a statistical model to accurately estimate post-FEC BER for high-speed wireline links using standard linear block codes, such as the RS(544,514,15) KP4 and RS(528,514,7) KR4 codes. A hierarchical approach is adopted to analyze the propagation of PAM-symbol and FEC-symbol errors through a two-layer Markov model. A series of techniques including state aggregation, time aggregation, state reduction, and dynamic programming are introduced making the time complexity to compute post-FEC BER below 10^{-15} reasonable. Error bounds associated with each method are found. The efficiency of the proposed model allows it to handle a larger state space, more DFE taps, and more sophisticated linear block codes than prior work. A 4-PAM 60 Gb/s wireline transceiver fabricated in a 7 nm FinFET technology is used as a test vehicle to validate this model. Measured data with two different channels reveals that the statistical model can properly predict the post-FEC error floor with standard FEC codes. While this paper demonstrates the method for capturing DFE error propagation, the method is general and can be applied to model other communication systems having memory effects. Moreover, our proposed model can be easily extended to higher-level PAM schemes and other advanced equalizer architectures to assist in making architectural choices for wireline transceivers.**

***Index Terms*— BER estimation; burst error; decision feedback equalization (DFE); dynamic programming; error propagation; forward error correction (FEC); linear block code; Markov model; pulse amplitude modulation (PAM); state aggregation; time aggregation; wireline channel.**

I. Introduction

FORWARD error correction (FEC) has become an integral part of many wireline links at data rates above 25 Gb/s whose impact must be considered when architecting transceivers to achieve a target BER below 10^{-15} without expensive overdesign [1]. A typical design practice, sometimes referred to as the FEC limit paradigm [2], is to design the serializer-deserializer (SerDes) for a targeted BER (e.g. 10^{-6}) without FEC, called the pre-FEC BER, assuming that an appropriate FEC code will correct most of the resulting errors providing a post-FEC BER of some desired level (e.g. 10^{-12} or 10^{-15}). However, this approach is naïve. For example, the 100GBase-KP4 standard [3] specifies transmitting 4-PAM symbols at 100 Gb/s over four backplane interconnects with less than 33 dB insertion loss at 7 GHz, targeting at a post-FEC BER better than or equal to 10^{-12} using a RS(544,514,15) FEC code. Depending on the equalization techniques used in the SerDes, the same pre-FEC BER may result in different post-FEC BER. In particular, error propagation in decision feedback equalization (DFE) can significantly impact BER. A DFE removes channel ISI by registering past equalized symbols in the feedback path and using them to estimate and cancel ISI from the current symbol. However, if any past decision registered in the DFE is wrong, the receiver's decision is biased and may increase the probability of additional symbol errors. Errors may thus propagate around the DFE feedback loop and result in FEC code failures. Unfortunately, simulations of the targeted post-FEC BERs are prohibitively long, especially for exploring architectural alternatives. Instead of using the FEC limit paradigm currently employed by many designs [4]–[6], which doesn't consider DFE error propagation, a model that accurately predicts very low post-FEC BERs is important for modern SerDes design.

Subject to various noise sources in wireline links [5], [7], [8], several models have been developed for BER estimation, each having its own limits. For example, the Gilbert model [9], [10] captures DFE burst errors, but its complexity grows exponentially with the number of DFE taps. Peak distortion analysis [11] focuses on the impact of residual (unequalized) inter-symbol interference (ISI) but may require too much simulation time to find all critical data patterns that contribute to BER.

A key challenge for statistical modeling is to accurately capture the impact of DFE error propagation on post-FEC BER. Ref [12] explains the approach in the IEEE 10GBASE standard for handling DFE error propagation. It considers bursts combining correct and erred bits, and enumerates all possible burst-error patterns to estimate BER and link performance.

Manuscript received June 14, 2019; revised August 29, 2019; accepted September 20, 2019. Date of publication October 22, 2019; date of current version January 15, 2020. This work was supported in part by the Le Fonds de recherche du Québec - Nature et technologies (FRQNT) under Grant 208736 and in part by the Natural Sciences and Engineering Research Council of Canada (NSERC) under Grant CRDPJ 505827-16 and Grant CGSD2-518889. This article was recommended by Associate Editor M. Onabajo. *(Corresponding author: Ming Yang.)*

M. Yang and A. Chan Carusone are with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4, Canada (e-mail: ming.yang@isl.utoronto.ca; tony.chan.carusone@isl.utoronto.ca).

S. Shahramian, H. Shakiba, H. Wong, and P. Krotnev are with Huawei Technologies Canada, Ottawa, ON K2K 3J1, Canada (e-mail: shayan.shahramian@huawei.com; hossein.shakiba@huawei.com; henry.wong@huawei.com; peter.krotnev@huawei.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSI.2019.2943569

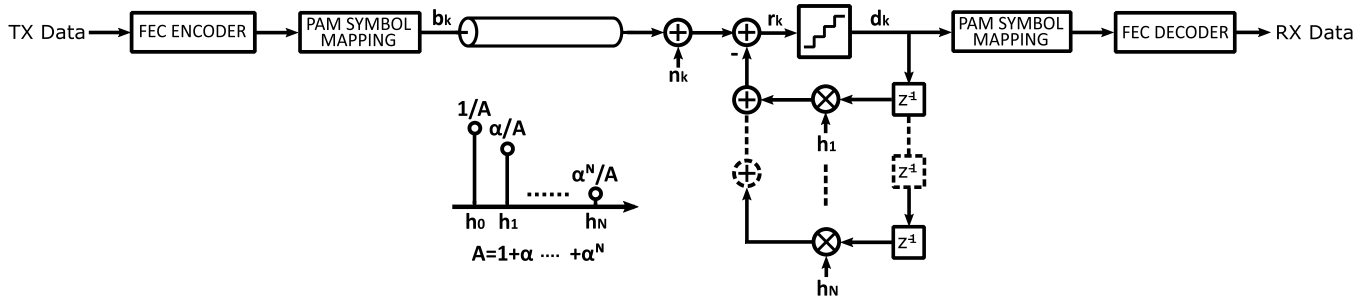


Fig. 1. A zero-forcing N-tap DFE example for wireline SerDes.

However, this time-consuming approach is ill-suited to the longer linear block codes adopted in recent wireline standards [13], [14]. Another possibility is to extrapolate to very low BERs based on a few simulations at higher BER [15]. The validity of such methods holds only if the BER-SNR correlation remains stable when BER is extrapolated to lower orders of magnitude, which we will show is impractical for many wireline links.

Past work on post-FEC BER estimation has focused on systems with BCH codes which operate in GF(2) using 2-PAM signaling [16]. A Markov chain model from [17] was adopted in [16] to account for DFE error propagation, and possible burst-error patterns are systematically grouped through trellis-based dynamic programming. However, 4-PAM signaling is becoming increasingly critical for 50 Gb/s+ wireline links [18]–[20], often with DFEs [21], [22]. Hence more powerful Reed-Solomon (RS) codes are being used to correct up to t FEC symbol errors caused by DFE error propagation. Nonetheless, very few attempts have been made to model and analyze the post-FEC BER for codes in higher-order Galois fields, GF(2^m), $m > 1$ in the presence of DFE error propagation [23], [24]. In [23], DFE error propagation across PAM symbols is considered using a method similar to [12], and post-FEC BER is calculated by enumerating all symbol-error combinations that result in $t+1$ or more FEC-symbol errors. It applied the method to only a 1-tap DFE, but the method's complexity can grow exponentially for a multi-tap DFE and large t values. In [24], the probability of having an error-free RS symbol is assumed to be independent of other symbol errors in a codeword, which may not be a valid assumption and thus incapable of accurately modeling error bursting for a post-FEC BER below 10^{-15}.

Our proposed BER estimation method for wireline links is an extension of [16], and provides a set of tools to assist in making architectural choices for wireline transceivers, such as co-design of the equalization and FEC in the presence of DFE error propagation and various noise sources. An extension from 2-PAM to Gray-Coded 4-PAM signaling is included in our model to calculate the bit-error probability for a multi-bit PAM symbol accurately. We generalize trellis-based dynamic programming to the FEC-symbol level, resulting in a hierarchical model containing many PAM sub-trellises allowing us to look at post-FEC BER for both non-interleaved and interleaved FEC codes in a reasonable amount of time. The model is simplified through state reductions to accelerate the statistical analysis. The efficiency of the proposed model allows it to handle a larger state space, more DFE taps, and more sophisticated linear block codes than prior work. Our proposed model can be easily extended to higher-level PAM schemes, and is also applicable to other advanced equalizer architectures that are likely to arise in ADC-based receivers for 100 Gb/s+ wireline links [25].

The modeling of DFE error propagation will be discussed in Section II. This will then be followed by the application of state aggregation and trellis-based dynamic programming to improve the computational efficiency of BER estimation in Section III. In Section IV we will propose a statistical model to estimate post-FEC BER for high-order PAM schemes and linear block FEC codes on GF(2^m), $m > 1$. A time-aggregated trellis model will be used to consider the error propagation at both the PAM-symbol and FEC-symbol levels. Section V will describe a method for post-FEC BER estimation and steps to minimize its computational complexity. Subsequently, in Section VI, the statistical model is experimentally verified on a 4-PAM 60 Gb/s SerDes link. Finally, conclusions are drawn in Section VII.

II. Modeling DFE Error Propagation

The statistical model proposed in [16] will be introduced in this section to estimate the pre-FEC BER in the presence of DFE error propagation. We first explain how DFE error propagation is modeled using Markov chain theory, and then apply trellis-based dynamic programming to efficiently collect probabilities of all error patterns that are needed for post-FEC BER calculation in Section IV.

First, consider the link model shown in Fig. 1, communicating symbols b_k with time index k. The symbols are filtered by a finite-impulse-response (FIR) channel response h_p with main cursor h_0, and subject to additive noise, n_k. Without limiting the scope of this work, it is assumed that all pre-cursor and higher-order post-cursor ISIs have been removed by linear equalizers. The detected symbols d_k may differ from the transmitted symbols resulting in the error sequence,

$$D_k = d_k - b_k. \tag{1}$$

This results in an additive error n_k^{dfe} generated by non-zero error terms in the DFE feedback path. Assuming a perfect zero-forcing N-tap DFE,

$$n_k^{dfe} = -\sum_{p=1}^{N} D_{k-p} h_p. \tag{2}$$

Then the DFE slicer input r_k becomes

$$r_k = b_k h_0 + n_k + n_k^{dfe}. \tag{3}$$

Error propagation is modeled as a Markov process whose state is specified by the error terms in the DFE feedback, $D_{k-1}, D_{k-2}, \ldots$ Assuming additive white Gaussian noise (AWGN) $n_k \sim \mathrm{N}(0, \sigma^2)$, we have $r_k \sim \mathrm{N}(b_k h_0 + n_k^{dfe}, \sigma^2)$. Hence, the rates at which $d_k \neq b_k$ and $d_k = b_k$ can be determined from the appropriate standard error function. This may be straightforwardly extended to include other impairments such as jitter, crosstalk, or residual ISI by appropriately changing the probability density function (pdf) of the received samples r_k [7], [8]. The one-step state-transition probabilities $q_{i'i}$ from a source state 'i'' to a sink state 'i' can be calculated by applying (3) to each pair of valid transitions $i'i$ in the Markov model, where the term n_k^{dfe} in (3) is exclusively dictated by the source state 'i''. With all $q_{i'i}$ calculated, we may find the steady-state probability, π_i, of any state i in the Markov model by solving the global balance equation [26],

$$\pi_i = \sum_{i'} q_{i'i}\pi_{i'}. \tag{4}$$

subject to

$$\sum_i \pi_i = 1. \tag{5}$$

III. Reducing Computational Complexity in Markov Model

A. Aggregation of Weakly Lumpable Markov Process

Applying state lumping (sometimes referred as state aggregation) to a Markov process allows the generation of an aggregated chain with a comparatively smaller state space resulting in reduced analytical complexity. The aggregated chain provides a coarser analysis of the state space and can be used to perform DFE error-rate analysis for the original Markov chain without losing analytical accuracy [27]. Consider a homogeneous and irreducible Markov process X with finite state space $S = \{1, 2, \ldots, s\}$ whose chain is defined by its one-step transition matrix $Q = [q_{i'i}]$ and initial probability vector γ. We say X is lumpable with respect to a partition $C = \{C_1, C_2, \ldots C_r\}$ given $\cup C_i = S$, $C_i \neq \emptyset$ and $C_u \cap C_v = \emptyset$ for any $u \neq v$ if the aggregated chain Y with state space $\bar{S} = \{1, 2, \ldots, r\}$ is also a homogeneous Markov process [28]. If the above definition holds true for all γ, we say X is strongly lumpable with respect to the partition C; if the above definition applies to at least one but not necessarily all choices of γ, then we say X is weakly lumpable with respect to C.

For an N-tap DFE with 2-PAM signaling, the original state space $S = \{1, 2, \ldots, 3^N\}$ can be reduced to $\bar{S} = \{1, 2, \ldots, 2^N\}$ using weak lumpability. A 2-tap DFE example is given in Fig. 2, and states are labelled according to the errors registered in the DFE: i.e. $< D_{k-1}, D_{k-2} >$. With 2-PAM $b_k = \pm 1$, $D_k \in \{+2, -2, 0\}$ and the DFE may be in $3^2 = 9$ different states as in Fig. 2(a). We obtain the $2^2 = 4$ Markov states in Fig. 2 (b) by lumping all +2 and -2 states at each DFE tap position. The lumped state ± 2 preserves the coarser bit-error information by discarding the sign of error value D_k.

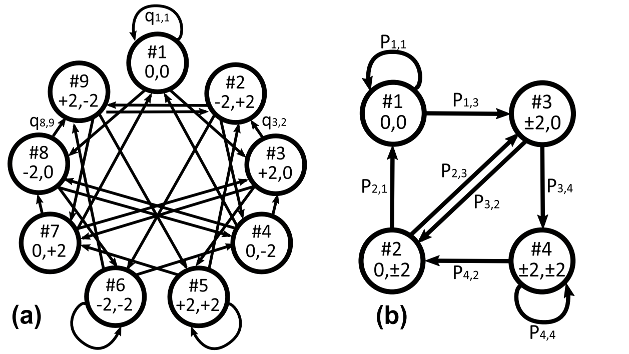


Fig. 2. Markov chain model for a 2-tap DFE and 2-PAM symbols $b_k \in \{-1, +1\}$: (a) before lumping [16] (b) after lumping. States are labelled D_{k-1}, D_{k-2}.

In the scope of this work, we consider the link illustrated in Fig. 1 subject to AWGN, having equally spaced DFE slicer thresholds, and an equally probable symbol set b_k that is independent of noise sample n_k. Under this particular setting, it is proven in [27] that, by exploiting the symmetry in the error states $< D_{k-1}, D_{k-2} \ldots, D_{k-N} >$, an N-tap DFE Markov process is weakly lumpable with respect to the partition lumping all states having the same error magnitude $|D_k|$ at each DFE tap. In addition, it is always assumed in our work that a Markov chain is initialized by its steady-state probability vector π, which is proven in [29] to be always a choice of γ leading to a homogeneous Markov chain if the chain is weakly lumpable.

In order to obtain the aggregated Markov model from the original one, we define γ^{Ci} as the restricted initial vector to a set C_i in partition C. For all elements in γ^{Ci}, we assign zeros to those that correspond to states not in C_i and normalize γ^{Ci} to a unit-sized vector. Therefore, the k^{th} element γ_k^{Ci} is

$$\gamma_k^{C_i} = \begin{cases} \frac{\gamma_k}{\sum_{j \in C_i} \gamma_j} \; if\ k \in C_i \\ 0 \; if\ k \notin C_i. \end{cases} \tag{6}$$

Let U_π be the $s \times r$ distributor matrix whose i^{th} row is π^{Ci}, which is the steady-state probability vector restricted to set C_i; let V be the $r \times s$ collector matrix generated by transposing the distributor matrix U_π and replacing all non-zero elements by 1. Denote $p_{i'mim}$ as the one-step state-transition probability from a lumped state 'i'_m' to a lumped state 'i_m'. Transition matrix $P = [p_{i'mim}]$ of the lumped process Y is then given by

$$P = U_\pi Q V. \tag{7}$$

Moreover, a more straightforward two-step procedure for computing the aggregated state probabilities from the original chain is provided in [30]. First, the aggregated steady-state probabilities Π_{im} can be calculated from the results obtained by (4) and (5),

$$\Pi_{i_m} = \sum_{i \epsilon i_m} \pi_i. \tag{8}$$

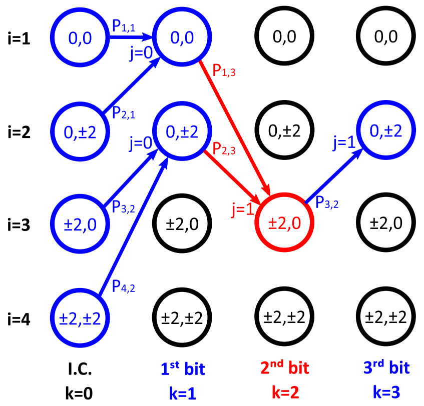


Fig. 3. 2-PAM trellis paths for having bit error at 2nd stage with $N = 2$ and $B = 3$.

Next, the aggregated state-transition probabilities $p_{i'mim}$ can be computed by

$$p_{i'_m i_m} = \sum_{i' \epsilon i'_m, i \epsilon i_m} \frac{\pi_{i'} q_{i' i}}{\Pi_{i'_m}}. \tag{9}$$

We may also numerically verify the weak lumpability using a sufficient condition proposed in [28]. That is, a Markov process X is weakly lumpable with respect to a partition C if

$$U_\pi Q V U_\pi = U_\pi Q. \tag{10}$$

B. Trellis-Based Dynamic Programming

We next apply trellis-based dynamic programming [31] to the Markov model to efficiently calculate the probability of bit errors in a codeword. The lumped Markov model for an N-tap DFE with M-PAM signaling may be represented by an M^N-state radix-M trellis. Rather than finding the BER by enumerating all possible error patterns in the trellis, dynamic programming solves the problem much faster by grouping the probability of all trellis paths having the same number of bit errors. The same aggregation procedure is repeated recursively when traversing through each stage in the trellis, resulting in a significant reduction in computational complexity. For a length-B, t-error-correcting block code, without dynamic programming, one must calculate the probability of all paths through the length-B trellis corresponding to $t + 1$ or more bit errors, adding them together to find the probability of a codeword error. For example, a trellis representation of the binary 2-tap DFE Markov model with $B = 3$ is shown in Fig. 3, highlighting all paths that result in exactly 1 detection error in the highlighted 2nd bit position. Combining the computed steady-state error probabilities and branch probabilities, one may compute the probability of each of these paths, along with those of paths having errors in the 1st bit and 3rd bit to find the total probability of 1-bit error within a 3-bit sequence. Unfortunately, the challenge of enumerating and computing these probabilities grows exponentially with block length B making the computations intractable for practical FEC codes typically having $B > 1{,}000$.

Instead, dynamic programming calculates the probability of long error patterns recursively in terms of state and error probabilities at the preceding stage. We denote $\Pr_k^j(i)$ the probability of arriving at Markov state i at time step k after traversing all trellis paths containing exactly j bit errors. For example, $Pr_2^1(3)$ represents the probability of arriving at state #3 at the 2nd stage of the trellis having traversed all paths corresponding to exactly 1 error. Hence, the bit-error probabilities at time $k + 1$, $Pr_{k+1}^j(i)$, can be found iteratively from the values of $Pr_k^j(i')$, $Pr_k^{j-1}(i')$ and the branch probabilities $p_{i'i}$. For example, for states 'i' where the most recently received bit is correct,

$$Pr_{k+1}^j(i) = \sum_{i'} Pr_k^j\left(i'\right) p_{i'i}. \tag{11}$$

whereas for states 'i' where the most recently received bit is incorrect,

$$Pr_{k+1}^j(i) = \sum_{i'} Pr_k^{j-1}\left(i'\right) p_{i'i}. \tag{12}$$

For example, according to (12), $Pr_2^1(3) = Pr_1^0(1)p_{13} + Pr_1^0(2)p_{23}$, corresponding to the red-highlighted paths in Fig. 3 where #1 and #2 are the only possible source nodes. Moreover, $Pr_1^0(1)$ and $Pr_1^0(2)$ can be found recursively from (11) when calculating all node probabilities for the 1st trellis stage. By repeating this procedure for all k, j and i, we will be able to obtain the probability of all error counts through the trellis with a computing time that increases only linearly with B. The recursion is initialized assuming the link has reached a steady-state, so $Pr_0^0(i) = \Pi_i$.

IV. 4-PAM Statistical Model for Non-Binary Linear Block Codes

In the previous section, we have reviewed a 2-PAM statistical model and the application of trellis-based dynamic programming to model DFE error propagation. In current long-reach wireline SerDes applications, such as the 100GBase-KP4, Gray-coded 4-PAM signaling and RS FEC are standard. For linear FEC codes on GF(2^m), the encoder groups every m bits into one FEC symbol, and correspondingly the decoder can detect and correct up to t erroneous FEC symbols in an n-symbol codeword. All m bit errors in each erred FEC symbol can be corrected so long as the total number of FEC symbol errors does not exceed t. Hence the higher-order RS codes provide stronger burst-error correction ability than BCH codes, a measure taken in part to accommodate DFE error propagation. In this section, we extend this statistical model to higher-order M-PAM schemes and linear block FEC codes on GF(2^m), for m being multiple integer of $\log_2(M)$ including the standardized wireline RS codes [13], [14]. The analysis is performed in two layers:

- First, a PAM trellis is defined to model the propagation of 4-PAM (physical-layer) symbol errors within a DFE over the course of one individual GF(2^m) FEC symbol.
- Second, a higher layer of analysis groups 4-PAM symbols into GF(2^m) FEC symbols through a time-aggregation approach and the probability of error propagation across

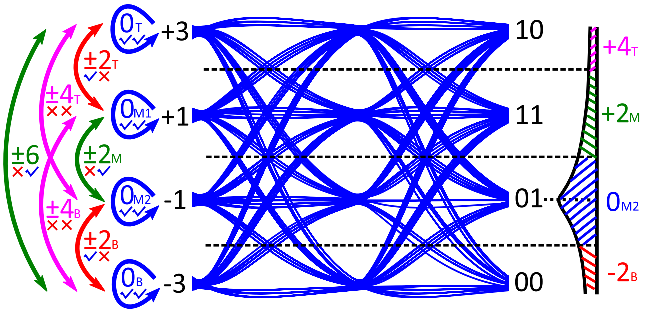


Fig. 4. A receiver eye diagram indicating all possible symbol-detection outcomes for a link communicating Gray-coded 4-PAM symbols $b_k \in \{\pm 3, \pm 1\}$.

FEC symbol boundaries is considered using a higher-level FEC trellis.

Dynamic programming is applied to analyze both trellises, ultimately resulting in the post-FEC BER.

A. 4-PAM Markov Model

With M-PAM signaling, there are in total M^2 symbol-detection outcomes considering all possible pairs of transmitted/detected PAM symbols. Hence an M-PAM N-tap DFE can be represented by an M^{2N}-state Markov model without applying state aggregation. Fig. 4 demonstrates a receiver eye diagram indicating all possible detection outcomes for a link communicating Gray-coded 4-PAM symbols $b_k \in \{\pm 3, \pm 1\}$. All 16 error values $D_k \in \{0_{\mathrm{T}}, 0_{\mathrm{M1}}, 0_{\mathrm{M2}}, 0_{\mathrm{B}}, \pm 2_{\mathrm{T}}, \pm 2_{\mathrm{M}}, \pm 2_{\mathrm{B}}, \pm 4_{\mathrm{T}}, \pm 4_{\mathrm{B}}, \pm 6\}$, together with their associated bit-error patterns, are also labeled in the same figure. The subscript of each error value denotes its relative position in the 4-PAM eye from top to bottom. Note that states having the same error value may correspond to different bit-error patterns. For example, subject to an error event $D_k = +2_{\mathrm{M}}$, the 1st bit of the received PAM symbol is in error, which corresponds to the pdf plot superimposed in Fig. 4 with $b_k = -1$, $d_k = +1$ and $n_k^{dfe} = 0$. However, the combination of $b_k = +1$ and $d_k = +3$ results in $D_k = +2_{\mathrm{T}}$, which instead makes the 2nd bit erroneous while having the same error value.

Next, in the M^{2N}-state Markov model, all states having the same error magnitude are aggregated together by applying weak lumpability, resulting in a much smaller M^N-state state space. Specifically, we can define a new set of $D_k \in \{0, \pm 2, \pm 4, \pm 6\}$ for the 4-PAM example given in Fig. 4. Steady-state and state-transition probabilities of the new aggregated chain can be calculated using (7)-(9), similar to what has been done in the 2-PAM case.

B. 4-PAM Trellis Model

When traversing in an M-PAM trellis using dynamic programming, each branch decision corresponds to between 0 and at most $\log_2 M$ bit errors. We define j_{PAM} as the number of bit errors in a PAM symbol detection. For example, in a link communicating 4-PAM symbols $b_k \in \{\pm 3, \pm 1\}$, $j_{PAM} \in \{0, 1, 2\}$ and the receiver error sequence defined in (1) is $D_k \in \{\pm 6, \pm 4, \pm 2, 0\}$. Assuming Gray-coding, an error value ± 2 or ± 6 corresponds to $j_{PAM} = 1$, whereas an error value ± 4 indicates $j_{PAM} = 2$. In each trellis iteration, for states 'i' where the most recently received 4-PAM symbol has j_{PAM}-bit errors,

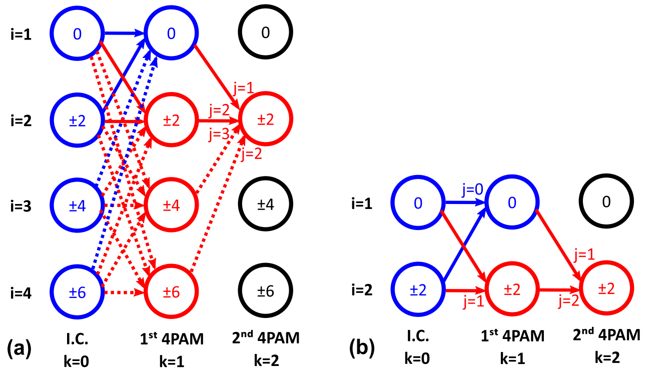


Fig. 5. 4-PAM trellis paths for calculating $\sum_j Pr_2^j(2)$ with $N = 1$ and $B = 2$ using (a) lumped trellis model (b) lumped trellis model ignoring ±4 and ±6 error events.

$$Pr_{k+1}^{j}(i) = \sum_{i'} Pr_k^{j-j_{PAM}}\left(i'\right) p_{i'i}. \qquad (13)$$

Fig. 5(a) shows an example for a 4-PAM 1-tap-DFE Markov model with $B = 2$, highlighting all possible paths ending in state ± 2 ($i = 2$). For example, $Pr_2^j(2)$ represents the probability of arriving at state #2 at the 2nd stage of the trellis having traversed all trellis paths corresponding to exactly j-bit errors, and the highlighted paths in Fig. 5(a) indicate all possible error patterns for calculating $\sum_j Pr_2^j(2)$. Hence, from (13) we know $\sum_j Pr_2^j(2) = Pr_1^0(1)p_{12} + Pr_1^1(2)p_{22} + Pr_1^2(3)p_{32} + Pr_1^1(4)p_{42}$, where the only possible node for $k = 1$ and $j = 2$ is #3. Without lumping the Markov model would have $7^1 = 7$ states for a 4-PAM 1-tap DFE, but it can be reduced to 4 as in Fig. 5(a) by lumping the 1-bit errors ±2/±6 and the 2-bit errors ±4. Note that lumping reduces the model's complexity much more as the number of DFE taps increases.

The trellis model can be further simplified to a 2^N-state radix-2 trellis as demonstrated in Fig. 5(b) by ignoring all the dotted paths in Fig. 5(a) that have unlikely ±4 and ±6 error events. In the following subsection, we will provide a quantitative justification and discuss the general conditions for ignoring these error events in the post-FEC BER analysis.

With the M-PAM trellis properly defined, a length $B = m/\log_2 M$ trellis may be analyzed using the methods in this section to find the probability of at least 1-bit error corrupting the GF(2^m) FEC symbol.

C. 4-PAM Trellis Model Simplification

We apply the 4-PAM Markov model to a link communicating $b_k \in \{\pm 3, \pm 1\}$ as depicted in Fig. 1 with $N = 4$ and channel response $1/A + \alpha/A\ z^{-1} + \alpha^2/A\ z^{-2} + \alpha^3/A\ z^{-3} + \alpha^4/A\ z^{-4}$. The channel response is normalized by $A = 1 + \alpha + \alpha^2 + \alpha^3 + \alpha^4$ to maintain a peak-amplitude constraint on the transmitter, typically imposed by supply voltage limitations. Hence, larger α indicates higher A, lower channel bandwidth, and a weaker main cursor in the channel response. A zero-forcing 4-tap DFE is assumed at the receiver and may thus

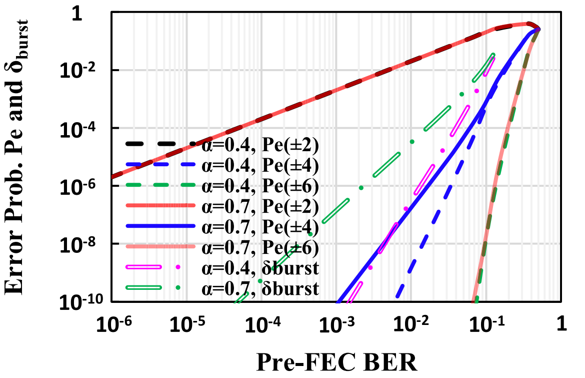


Fig. 6. Pre-FEC BER versus probability of ±2, ±4 and ±6 error events, and versus δ_{burst} with $N = 4$, $m = 10$, 4-PAM signaling and AWGN noise. The pre-FEC BER is obtained by calculating the weighted average of all error event probabilities using the 4-PAM Markov model.

introduce error propagation. As α is increased, a lower noise variance σ^2 is required to maintain the same pre-FEC BER and thus a larger proportion of errors are caused by DFE error propagation.

Fig. 6 plots the probability of each error value versus pre-FEC BER with $\alpha = 0.4$ and 0.7, respectively. Noise variance σ^2 is swept to generate each curve. Clearly, for each slicer decision the probability of ±2 error events (associated with the nearest-neighboring PAM signal levels) will be greater than ±4 and ±6 error events. Despite the very large DFE tap weights in these channels, the probability of ±4 and ±6 error events are several orders of magnitude lower than ±2 error events. This fact can be also qualitatively verified by the example given in Fig. 4, where the noise pdf is obtained by setting $b_k = -1$, $d_k = +1$ and $n_k^{dfe} = 0$. A smaller noise variance σ^2 leads to a tighter pdf distribution and a lower BER, and the area of each shaded pdf region is proportional to the probability of each error event. As BER decreases, the ±4 and ±6 event probabilities corresponding to the area under the Gaussian-like exponential tail declines much faster than the probability of ±2 error events. This ultimately results in a much higher slope for ±4 and ±6 events in the plot. As pre-FEC BER is the weighted average of these error event probabilities, neglecting ±4 and ±6 error events will not impact the accuracy of pre-FEC BER analysis at levels below 10^{-2}.

When traversing the PAM trellis in a codeword, all error patterns contributing to the post-FEC BER are recursively computed by aggregating the probability of all trellis paths having more than t FEC symbol errors. It is also possible to neglect ±4 and ±6 error events in post-FEC error-rate analysis if the probability of burst errors across multiple FEC symbols is not impacted. This can be quantitatively justified by analyzing the error propagation probability P_{burst} between two neighboring FEC symbols using a $2m/\log_2 M$-stage M-PAM trellis, where every $m/\log_2 M$ stages represents one FEC symbol in GF(2^m). We denote $Pr[x]_k^j(i)$ as the probability of arriving at Markov state i at time step k after traversing all trellis paths containing exactly j bit errors in the x^{th} FEC symbol. With an M^N-state Markov model, we have $i \in [1, M^N]$, $j \in [0, k{\cdot}\log_2 M]$, $k \in [0, m/\log_2 M]$ and $x \in [1, 2]$ in each trellis iteration. We can obtain P_{burst} by calculating the error probability of a FEC symbol given an erroneous preceding FEC symbol. First, we traverse the trellis for $x = 1$. Then, the probability space in the leading FEC symbol is normalized by excluding all error-free trellis paths using scaling factor

$$c = \sum_i \sum_{j>0} Pr[1]^j_{m/log_2 M}(i). \tag{14}$$

Next, the normalized probability of visiting state i at the last PAM stage in the erred leading FEC symbol, becomes the initial condition $Pr[x+1]_0^0(i)$ of the following FEC symbol,

$$Pr[x+1]_0^0(i) = \sum_{j>0} Pr[x]^j_{m/log_2 M}(i)/c. \tag{15}$$

Similarly, we can use the above method to generate P'_{burst} by ignoring the ±4 and ±6 error events. The relative error introduced by the simplified trellis model is

$$\delta_{burst} = \left| \frac{P_{burst} - P'_{burst}}{P_{burst}} \right|. \tag{16}$$

With $m = 10$, $M = 4$ and $N = 4$, δ_{burst} versus pre-FEC BER for $\alpha = 0.4$ and 0.7 is also reported in Fig. 6. For the case where $\alpha = 0.7$, a larger proportion of errors are caused by DFE error propagation which increases the probability of ±4 and ±6 error events. The relative error δ_{burst} monotonically decreases with smaller σ^2, and $\delta_{burst} \leq 0.01\%$ for pre-FEC BER $\leq 10^{-2}$. When estimating the probability of a codeword containing 100 FEC symbol errors, the relative estimation error is bounded by the worst-case scenario having 100 consecutive errors, 1-(1-0.0001)$^{100} \approx 1\%$. As modern SerDes links generally operate with a pre-FEC BER $\leq 10^{-2}$, $m = 10$, $N \leq 4$, $t = 15 \ll 100$, and $\alpha \leq 0.4$ [19]–[21], the simplified trellis model can be practically applied to the post-FEC error-rate analysis with an estimation error much less than 1%. In addition, we can always apply the original 4^N-state PAM trellis to verify the results generated by the simplified model. Therefore, to further reduce the complexity of the model, we consider a 2^N-state radix-2 trellis for all 4-PAM analysis with $D_k \in \{\pm 2, 0\}$ in the remainder of this work.

D. Time-Aggregated FEC Trellis Model

Using the methods described so far, every FEC symbol in GF(2^m) can be decomposed into a length-$m/2$ 4-PAM trellis describing link behavior in the physical layer. Recall the example in Fig. 5 that we apply (13) to recursively compute $\text{Pr}_k^j(i)$ in order to aggregate the probability of error patterns having exactly j bit errors, where $j \in \{0 \ldots m/2\}$.[a] With an N-tap DFE, this requires a total number of

$$\sum\nolimits_{k=1}^{m/2} \sum\nolimits_{j=0}^{k} 2^N = O\left(2^N \cdot m^2\right) \tag{17}$$

iterations to analyze the probability of all error patterns in a length-$m/2$ 4-PAM trellis.

[a] The possibility of more than $m/2$ bit errors in m 4-PAM symbols is ignored as per section IV-C.

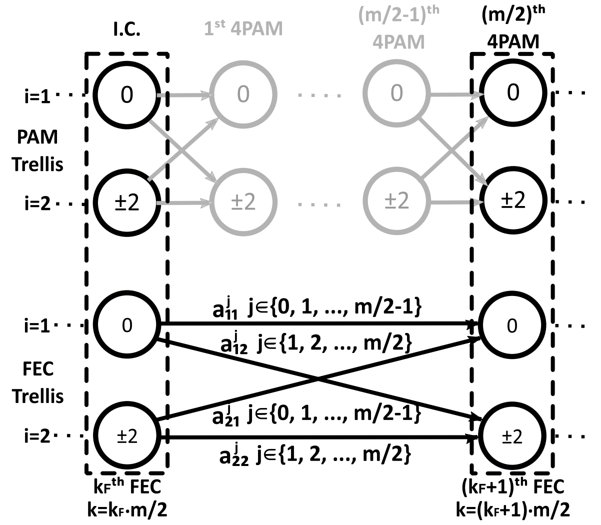


Fig. 7. A time-aggregated 4-PAM trellis example with $N = 1$.

Note that all paths in the trellis representing $Pr_k^j(i)$, the probability of arriving at state i at the k^{th} stage of the trellis after traversing all trellis paths containing exactly j bit errors, can be decomposed into 2^N groups of trellis paths and each starts with one of the 2^N Markov states at $k = 0$. For example, in Fig. 5(b) all trellis paths representing $Pr_2^1(2)$ must begin with one of the two DFE states at $k = 0$. As such, we may simplify the entire length-$m/2$ 2^N-state radix-2 trellis to a length-1 2^N-state radix-($2^N{\cdot}m/2$) trellis by aggregating all j-bit-error paths within each of the 2^N groups to a one-step direct transition between the two states at $k = 0$ and $k = m/2$. Each one-step transition in the simplified trellis is equivalent to traversing $m/2$ 4-PAM symbols in the fully expanded trellis. Fig. 7 shows an example of a time-aggregated 4-PAM trellis with $N = 1$, where we denote $a_{i'i}^j$ as the one-step state-transition probability from source state 'i'' to sink state 'i' with exactly j bit errors. Depending on the choice of sink state 'i' and the number of aggregated PAM-symbol stages, there are in total $m/2$ possible transitions between any of the two states in the simplified trellis. For example, for the transition a_{22}^j in Fig. 7, $j \in \{1 \ldots m/2\}$ as all the aggregated paths end at $i = 2$ has at least 1 bit error.

As such, we may construct a new trellis model for the entire FEC block, assuming that each state transition from the k_F^{th} to the $(k_F{+}1)^{\text{th}}$ stage has traversed a group of length-$m/2$ PAM-trellis paths. This is referred as the time aggregation of a Markov decision processes [32]; we group trellis paths over $m/2$ consecutive 4-PAM symbols while the time-aggregated Markov model preserves both the time-homogeneity and bit-error information. We call this time-aggregated PAM trellis the FEC trellis model, distinguishing it from the PAM symbol-level trellis considered thus far. With this approach, a total number of

$$2^N \cdot 2^N \cdot m/2 = O\left(2^{2N} \cdot m\right) \tag{18}$$

iterations are required to analyze the probability of all error patterns in a FEC symbol. Compared with the computational complexity calculated in (17), the time-aggregation technique outperforms when $m > 2^N$. In current wireline FEC standards, both the RS(544, 514, 15) KP4 and RS(528, 514, 7) KR4 codes are in GF(2^{10}) [13], [14]. In addition, due to the trade-off between power, area, and speed in a multi-tap DFE design, $N \leq 2$ in most high-speed wireline applications [20]–[22]. Therefore, time aggregating the underlying PAM trellis of each FEC symbol results in a significant reduction in computational complexity.

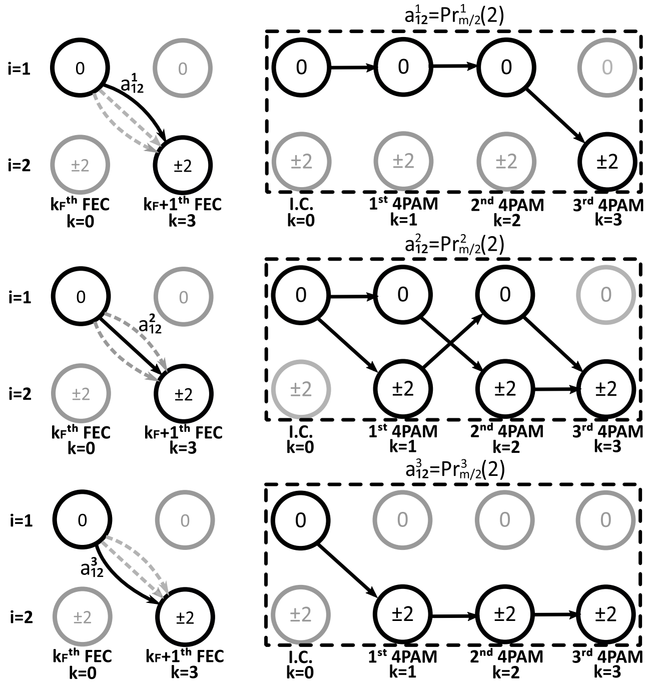


Fig. 8. Time aggregating a 4-PAM trellis with $m = 6$ and $N = 1$ showing the time-aggregated PAM trellis and the corresponding aggregated one-step state-transition probability in the fully expanded PAM trellis.

In order to analyze the FEC trellis, we must first find all the state-transition probabilities of these 2^N states by analysis of each underlying 4-PAM trellis. Fig. 8 shows an example illustrating the time-aggregation of a 4-PAM trellis for $N = 1$ and $m = 6$. The FEC trellis is expanded in Fig. 8 showing the underlying 4-PAM trellis to illustrate how we may find state-transition probabilities $a_{i'i}^j$ in the FEC trellis. First, we instantiate the expanded PAM trellis by assuming that the PAM trellis starts at the state 'i'' in $a_{i'i}^j$ with a probability of 1,

$$Pr_0^0\left(i'\right) = 1. \tag{19}$$

Next, after traversing the expanded 4-PAM trellis using the dynamic programming procedure described in (13), the transition probability $a_{i'i}^j$ to the next $(k_F + 1)^{\text{th}}$ FEC trellis stage can be calculated by summing the probability of all j-bit-error PAM-trellis paths ending at state 'i',

$$a_{i'i}^j = Pr_{m/2}^j\left(i\right)\Big|_{Pr_0^0(i')=1}. \tag{20}$$

For example, in Fig.8, a_{12}^2 corresponds to the summed probability of all PAM-trellis paths starting with state $i = 1$ and ending at $i = 2$ where 2 bit errors are detected in the fully expanded PAM trellis. For this particular case,

$$a_{12}^2 = Pr_3^2(2)\Big|_{Pr_0^0(1)=1} = p_{11}p_{12}p_{22} + p_{12}p_{21}p_{12}. \quad (21)$$

In the FEC trellis, each transition is equivalent to traversing a length-$m/2$ 4-PAM trellis. The initial state probabilities are those at the last PAM stage of the previous FEC symbol. Thus, we can recursively compute $Pr_k^j(i)$ in a fully-expanded length-$m/2$ trellis, given the initial probability vector $\gamma = [Pr_0^0(1)\ldots Pr_0^0(2^N)]^{\mathrm{T}}$. By applying the time aggregation technique, the probability of error propagation across FEC symbols is captured by directly computing the $Pr_k^j(i)$ between FEC symbol boundaries. The probability of arriving at state i at the last $(m/2)^{\text{th}}$ 4-PAM stage in a FEC symbol after traversing all paths containing exactly j bit errors, is the sum of $a_{i'i}^j$ with respect to all possible source node 'i'' and weighted by the initial probability vector γ,

$$Pr_{m/2}^j(i) = \sum_{i'} Pr_0^0(i')a_{i'i}^j, \quad (22)$$

which proves the fact that the time-aggregation technique is equivalent to directly traversing the fully-expanded PAM trellis while the previous case could potentially benefit from a reduced computational complexity if $m > 2^N$. Note that all state-transition probabilities $a_{i'i}^j$ in the FEC trellis are independent of the time index k_F and initial probability vector γ, resulting in a stationary FEC Markov model where all $a_{i'i}^j$ only need to be computed once.

E. Dynamic Programming for FEC Codes in GF(2^m)

To compute the post-FEC BER, we must apply dynamic programming to enumerate the probability of all error patterns having more than t FEC symbol errors in a codeword. However, the dynamic programming algorithm described by (11-13) can only track the total number of bit errors. Therefore, we create another error index allowing us to aggregate all error patterns in terms of both FEC symbol errors and bit errors. In the FEC trellis, we denote $Pr\_FEC_{k_F}^{js,jb}(i)$ the probability of visiting Markov state i at time step k_F after traversing all trellis paths containing exactly j_s FEC symbol errors and j_b bit errors. Hence, the error probabilities at time $k_F + 1$, $Pr\_FEC_{k_F+1}^{js,jb}(i)$, can be found iteratively from the values of $Pr\_FEC_{k_F}^{js,jb}(i)$ and the branch probabilities $a_{i'i}^j$. For a transition to state 'i' in the FEC trellis where the traversed $m/2$ PAM symbols have exactly j bit errors,

$$Pr\_FEC_{k_F+1}^{j_s,j_b}(i) = \sum_{i'} Pr\_FEC_{k_F}^{j_s-min(1,j),j_b-j}\left(i'\right)a_{i'i}^j. \quad (23)$$

V. Post-FEC BER Estimation and Model Optimization

A. Post-FEC BER Estimation

We first define $Pr\_FEC_n^{js,jb}$ as the grouped probability of all error patterns having j_s symbol errors and j_b bit errors along with a FEC trellis path of length n, computed by

$$Pr\_FEC_n^{j_s,j_b} = \sum_i Pr\_FEC_n^{j_s,j_b}(i). \quad (24)$$

Next, denote $W(j_s)$ the probability of having exactly j_s FEC symbol errors in an n-symbol codeword,

$$W(j_s) = \sum_{j_b=j_s}^{j_s\cdot\frac{m}{2}} Pr\_FEC_n^{j_s,j_b}. \quad (25)$$

To calculate BER, we define $E_{avg}(j_s)$ as the average number of bit errors in each erroneous FEC symbol given that exactly j_s symbol errors occurred in an n-symbol codeword,

$$E_{avg}(j_s) = \frac{\sum_{j_b=j_s}^{j_s\cdot\frac{m}{2}}\left(Pr\_FEC_n^{j_s,j_b}\cdot j_b\right)}{j_s\cdot W(j_s)}. \quad (26)$$

Then, the pre-FEC BER can be calculated as

$$BER_{pre-FEC} = \sum_{j_s=1}^{n}\left[\frac{W(j_s)\cdot E_{avg}(j_s)\cdot j_s}{n\cdot m}\right]. \quad (27)$$

Finally, to estimate the post-FEC BER for a t-error correcting RS code in GF(2^m) of block length n,

$$BER_{post-FEC} = \sum_{j_s=t+1}^{n}\left[\frac{W(j_s)\cdot E_{avg}(j_s)\cdot j_s}{n\cdot m}\right]. \quad (28)$$

When evaluating BER, the time-complexity of the dynamic programming procedures may become excessive because all combinations of j_b and j_s must be iterated at each trellis stage. For an n-symbol codeword in GF(2^m), the 2^N-state FEC trellis model would require a total of

$$\sum\nolimits_{k_F=1}^{n}\sum\nolimits_{j_s=1}^{k_F}\sum\nolimits_{j_b=1}^{j_s\cdot\frac{m}{2}}\left(2^{2N}\cdot\frac{m}{2}\right) = O\left(2^{2N}\cdot m^2\cdot n^3\right) \quad (29)$$

iterations. In Section V-B, we will propose a pruning method to improve the analytical complexity of this dynamic-programming algorithm.

B. Pruning-Based Dynamic Programming Algorithm

At low BER, as $W(j_s)$ decreases exponentially with increasing j_s, pruning trellis paths having negligible probabilities can result in a significant reduction in computation. This is achieved by replacing the upper summation limit n in (27) and (28) with j_s^{max}, indicating only trellis paths having up to j_s^{max} FEC symbol errors are preserved. Selecting j_s^{max} is based on the accuracy requirement on post-FEC BER through an iterative algorithm that will be explained later. Hence the same FEC trellis model would require a total of

$$\sum\nolimits_{k_F=1}^{n}\sum\nolimits_{j_s=1}^{j_s{}^{max}}\sum\nolimits_{j_b=1}^{j_s\frac{m}{2}}\left(2^{2N}\cdot\frac{m}{2}\right) = O\left((j_s^{max})^2\cdot 2^{2N}\cdot m^2\cdot n\right) \quad (30)$$

iterations.

Consider the trellis tree diagram in Fig. 9 for $n = 4$. All trellis paths having $j_s > j_s^{max}$ (dashed lines) are discarded

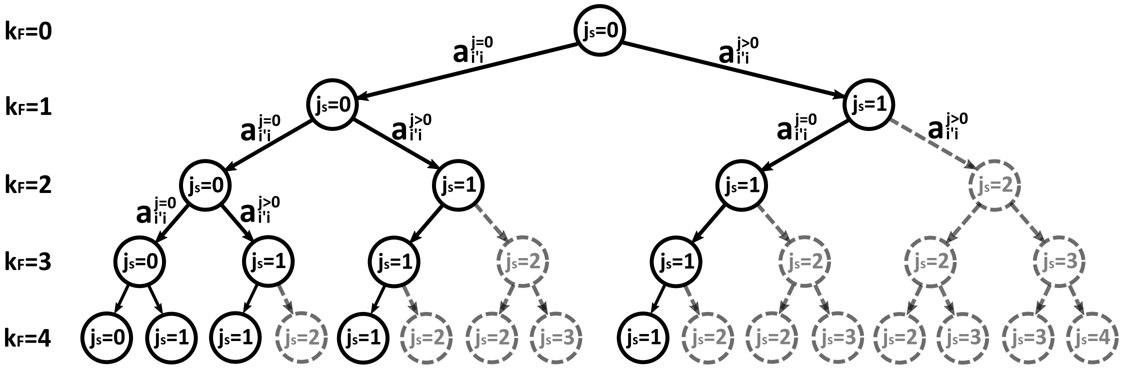


Fig. 9. Pruning FEC trellis paths with $j_s^{max} = 1$ in a 4-symbol codeword.

during each dynamic programming iteration by (23). First of all, the post-FEC BER can be calculated by modifying the upper summation limit in (28)

$$BER_{post-FEC} \approx \sum_{j_s=t+1}^{j_s^{max}} \left[\frac{W(j_s) \cdot E_{avg}(j_s) \cdot j_s}{n \cdot m} \right]. \quad (31)$$

Since all paths having more than j_s^{max} symbol errors are neglected, $W(j_s) = 0$ and $E_{avg}(j_s) = 0$ for $j_s > j_s^{max}$. Naturally, some error is incurred by neglecting the pruned paths, but we can accurately estimate this error to ensure it is negligible. We define $\varepsilon(j_s^{max})$ as the summed probability of all truncated paths,

$$\varepsilon\left(j_s^{max}\right) = \sum_{j_s=j_s^{max}+1}^{n} W(j_s) = 1 - \sum_{j_s=1}^{j_s^{max}} W(j_s). \quad (32)$$

Moreover, $\varepsilon(j_s^{max}) \approx W(j_s^{max}+1)$ since $W(j_s)$ decreases exponentially with increasing j_s. Consequently, the absolute error in the BER estimate of (31) can be approximated by

$$e\left(j_s^{max}\right) = \sum_{j_s=j_s^{max}+1}^{n} \left[\frac{W(j_s) \cdot E_{avg}(j_s) \cdot j_s}{n \cdot m} \right] \approx \frac{\varepsilon\left(j_s^{max}\right) \cdot E_{avg}\left(j_s^{max}+1\right) \cdot \left(j_s^{max}+1\right)}{n \cdot m}. \quad (33)$$

We may use the fact that $E_{avg}(j_s^{max}+1) \approx E_{avg}(j_s^{max})$ to approximate $e(j_s^{max})$ without having to calculate $E_{avg}(j_s^{max}+1)$ using the full FEC trellis. Moreover, for a FEC code correcting t symbol errors, we may also define the relative error $e_r(j_s^{max})$ in our estimate of post-FEC BER,

$$e_r\left(j_s^{max}\right) \approx \frac{\varepsilon\left(j_s^{max}\right) \cdot E_{avg}\left(j_s^{max}+1\right) \cdot \left(j_s^{max}+1\right)}{\sum_{j_s=t+1}^{j_s^{max}} \left[W(j_s) \cdot E_{avg}(j_s) \cdot j_s\right]}. \quad (34)$$

The negligibility of the effect of this pruning approach can be illustrated by an example demonstrated in Fig. 10. The statistical model is applied to the link depicted in Fig. 1 with $n = 544$, $t = 15$, $m = 10$ and $N = 4$ for two different channel α settings. Under the same 10^{-3} pre-FEC BER, results for $e_r(j_s^{max})$ and $W(j_s)$ are plotted in Fig. 10. A larger α intensifies DFE error propagation and thus results in increased $W(j_s)$ at longer burst lengths. Since the same pre-FEC BER is assumed in both cases, the $\alpha = 0.4$ channel has shorter bursts and thus higher $W(j_s)$ over shorter burst lengths. The relative error function $e_r(j_s^{max})$ also increases with a larger α but decreases exponentially by increasing j_s^{max}. Accurate pre-FEC BER results are obtained by computing the weighted average of all error event probabilities provided in Section IV-C. The best value of j_s^{max} can be determined by iterating from $j_s^{max} = t+1$ until a given accuracy requirement η on $e_r(j_s^{max})$ is met at a pre-selected pre-FEC BER level which corresponds to the desired post-FEC BER. For the example given in Fig. 10, if η is 1% with 10^{-3} pre-FEC BER, the best choice of j_s^{max} is 18 for both α settings.

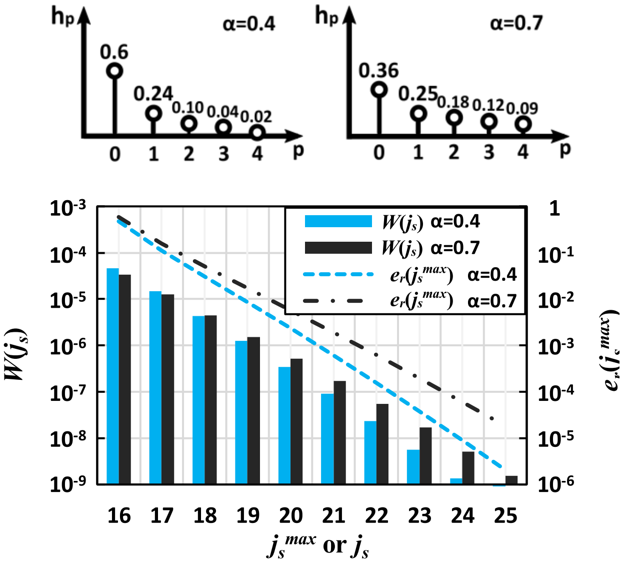


Fig. 10. $e_r(j_s^{max})$ and $W(j_s)$ for various channel α settings at 10^{-3} pre-FEC BER.

C. Model Verification

A 4-PAM statistical model is applied to a link as depicted in Fig. 1 with a channel response $h = 0.6 + 0.2z^{-1}$ - $0.2z^{-2}$. Such a response may, for example, arise from the combination of a lowpass channel and a continuous time linear equalizer (CTLE) that over-equalizes the channel. The solid line in Fig. 11 reports the pre-FEC vs post-FEC BER calculated

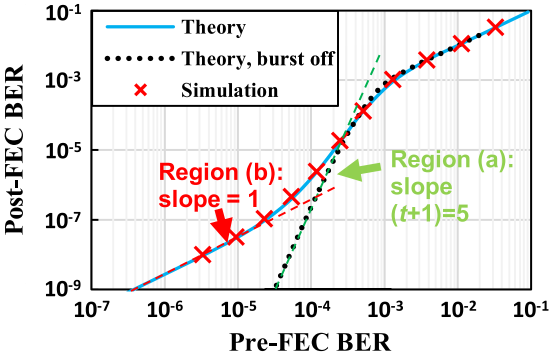


Fig. 11. Pre-FEC vs post-FEC BER plot for RS(544,536,4) with $h = 0.6 + 0.2z^{-1}$ - $0.2z^{-2}$.

using the methods described above with $j_s^{max} = 10$ for the RS(544,536,4) code on GF(2^{10}). The dotted line reports the results neglecting DFE burst errors. Behavioral simulation results are superimposed on the same axes to verify the correctness of our model down to a post-FEC BER of 10^{-8}.

In Fig. 11, we may identify two regions of interest. First, consider an extreme case where no burst errors are present. In such a case, a codeword will be decoded incorrectly only when there are (t+1) random bit errors, each having probability p. Hence, post-FEC BER $\sim p^{(t+1)}$. This case corresponds to the region (a) in Fig. 11, where the slope of Post-FEC vs. Pre-FEC BER is (t+1) on a logarithmic scale. Another extreme case can be represented by region (b), where individual random bit errors turning into very long bursts are the dominant source of post-FEC errors. If some small fraction, b, of pre-FEC random errors will generate bursts long enough to create post-FEC errors, post-FEC BER $\sim b{\cdot}p$. Thus, the slope of post-FEC vs. pre-FEC BER in this region is 1 on a logarithmic scale.

However, our statistical model does not consider decoder failures in the presence of more than t symbol errors, where the decoder may correct to the wrong codeword, thus increasing the number of bit errors. The probability of such a decoder failure is bounded by 1/t! [33]. In typical wireline SerDes applications, t is relatively large to correct burst errors, so that decoding to the wrong codeword does not affect the modeling accuracy of, for example, the standard RS(544,514,15) code.

VI. Experimental Verification

A. Device Under Test

We have measured a 4-PAM 60 Gb/s SerDes link based on a chip fabricated in 7 nm FinFET technology [34]. The overall system-level block diagram of the link is plotted in Fig. 12. Specifically, subject to a 1V$_{\text{ppd}}$ maximum output swing, the transmitter has a programmable 3-tap FIR filter to mitigate both pre-cursor and post-cursor ISI. At the receiver, a 13-tap FFE with 5 pre-cursor taps and 7 post-cursor taps is adaptively optimized to cancel ISIs in the channel. A 2-tap DFE equalizes the first two post-cursor ISIs. A statistical unit on-chip monitors and stores BER for PRBS31 data in memory. Both the RS(544, 514, 15) KP4 and RS(528, 514, 7) KR4 codes in GF(2^{10}) are implemented in the FEC encoder/decoder.

B. Modeling 2:1 Bit Multiplexing

To comply with IEEE wireline system standards, a 2:1 bit multiplexer is implemented in the PMA sublayer as illustrated in Fig. 12. The 2:1 bit multiplexing provides an extra layer of complication and must be considered in our proposed statistical model. Fig. 13 demonstrates an example showing FEC symbol distribution and 2:1 bit multiplexing at the transmitter. FEC symbols $C_1, C_2, \ldots C_{544}$ in a KP4-encoded codeword are distributed to two PCS lanes (in a round-robin fashion). Then, a bit multiplexer in the PMA layer groups every two bits from each PCS lane and forms a physical-layer 4-PAM symbol. At the receiver, the signal flow in Fig. 13 is reversed to retrieve the codeword C. As a result, burst errors in the physical layer are shuffled across multiple FEC symbols thus making the BER worse.

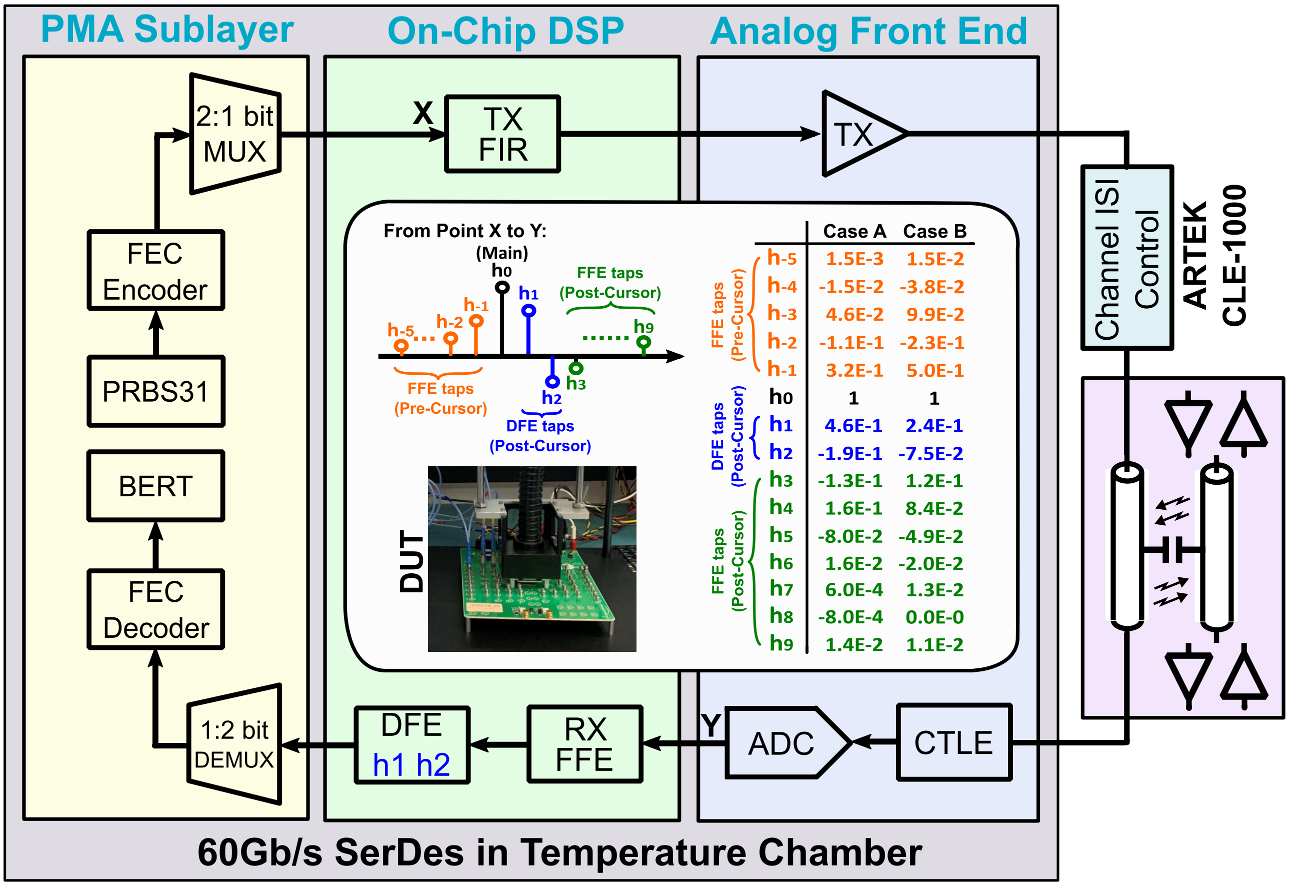
To model 2:1 bit multiplexing, we carefully consider the error pattern of each erroneous 4-PAM symbol and identify the exact bit-error location. First, we apply weak lumpability to define a new set of simplified 4-PAM error states. Whereas we previously lumped together all 4-PAM symbol errors with value ±2, we must now distinguish between errors in the first and second bit of the Gray-coded symbol. Thus, from the original 16 error values $D_k \in \{0_{\text{T}}, 0_{\text{M1}}, 0_{\text{M2}}, 0_{\text{B}}, \pm 2_{\text{T}}, \pm 2_{\text{M}}, \pm 2_{\text{B}}, \pm 4_{\text{T}}, \pm 4_{\text{B}}, \pm 6\}$ by ignoring ±4 and ±6 error events, the new DFE error states are $D_k \in \{0, \pm 2_{\text{MSB}}, \pm 2_{\text{LSB}}\}$, where the aggregated state $\pm 2_{\text{MSB}} = \{\pm 2_{\text{M}}\}$ and $\pm 2_{\text{LSB}} = \{\pm 2_{\text{T}}, \pm 2_{\text{B}}\}$ represent a first-bit error and a second-bit error, respectively. Hence, an N-tap DFE may be represented by a 3^N-state radix-3 4-PAM trellis. In each PAM trellis iteration, two bit-error indexes are needed so we are able to know exactly which of the two FEC symbols is affected by the erroneous bit. We denote $\Pr_k^{j1,j2}(i)$ the probability of arriving at Markov state i at time step k after traversing all PAM-trellis paths containing exactly j_1 MSB errors and j_2 LSB errors. For states 'i' where the most recently received 4-PAM symbol is $\pm 2_{\text{MSB}}$,

$$Pr_{k+1}^{j_1,j_2}(i) = \sum_{i'} Pr_k^{j_1-1,j_2}\left(i'\right) p_{i'i}. \tag{35}$$

For states 'i' where the most recently received 4-PAM symbol is $\pm 2_{\text{LSB}}$,

$$Pr_{k+1}^{j_1,j_2}(i) = \sum_{i'} Pr_k^{j_1,j_2-1}\left(i'\right) p_{i'i}. \tag{36}$$

Then, in the FEC trellis model, as the 2:1 bit multiplexing correlates every two FEC symbols in GF(2^{10}), trellis paths over every 10 consecutive 4-PAM symbols are time-aggregated to obtain our FEC trellis analysis of error propagation and RS FEC decoding. Hence, we consider each transition in the FEC trellis having traversed a length-10 4-PAM trellis with j_1 MSB errors and j_2 LSB errors. This results in a 3^N-state radix-($5{\cdot}3^N$) FEC trellis model if we neglect all ±4 and ±6 error events, where all the branch probabilities $a_{i'i}^{j1,j2}$ can be found using procedures described in section IV-D. To perform



		Case A	Case B
FFE taps (Pre-Cursor)	h_{-5}	1.5E-3	1.5E-2
	h_{-4}	-1.5E-2	-3.8E-2
	h_{-3}	4.6E-2	9.9E-2
	h_{-2}	-1.1E-1	-2.3E-1
	h_{-1}	3.2E-1	5.0E-1
	h_0	1	1
DFE taps (Post-Cursor)	h_1	4.6E-1	2.4E-1
	h_2	-1.9E-1	-7.5E-2
FFE taps (Post-Cursor)	h_3	-1.3E-1	1.2E-1
	h_4	1.6E-1	8.4E-2
	h_5	-8.0E-2	-4.9E-2
	h_6	1.6E-2	-2.0E-2
	h_7	6.0E-4	1.3E-2
	h_8	-8.0E-4	0.0E-0
	h_9	1.4E-2	1.1E-2

Fig. 12. System-level block diagram and test setup of the 60 Gb/s SerDes link [34].

dynamic programming on the FEC trellis, we still denote $Pr\_FEC^{j_s,j_b}_{k_F}(i)$ as the probability of visiting state i at time step k_F after traversing all trellis paths containing exactly j_s FEC symbol errors and j_b bit errors. For a transition to state 'i' in the FEC trellis where the traversed 10 PAM symbols have exactly j_1 MSB errors and j_2 LSB errors,

$$Pr\_FEC^{j_s,j_b}_{k_F+1}(i) = \sum_{i'} Pr\_FEC^{j_s-min(1,j_1)-min(1,j_2),j_b-j_1-j_2}_{k_F}\left(i'\right) a^{j_1,j_2}_{i'i}. \tag{37}$$

C. Test Setup

The test bench setup for the 60 Gb/s SerDes link is also superimposed in Fig. 12. A FlexTC temperature forcing system from Mechanical Devices is used to keep the device at room temperature with ±0.2 °C accuracy. Approximately Gaussian-distributed crosstalk noise is coupled to the channel through a crosstalk injection board. Different measurement cases are established by varying the channel insertion loss using an ARTEK CLE1000 variable ISI channel. The corresponding overall pulse responses (including TX FIR, TX driver, channel, RX CTLE and ADC) for two different cases are also tabulated in Fig. 12.

In case A, the overall insertion loss is 29 dB. We intentionally configure the CTLE in this case to over-equalize so that the second post-cursor ISI of the overall impulse response becomes large but negative. DFE error propagation is particularly bad in this case compared with all-positive post-cursor ISIs.[b] With large negative DFE tap weights, a measurable floor is expected in the post-FEC BER where burst errors due to error propagation in the DFE dominate. In this region, we expect to see a plot of post- vs. pre-FEC BER exhibit a slope of 1. In case B, the system has a lower overall insertion loss of 24 dB so that the KR4 code can provide adequate coding gain at low BER.

D. Experimental Results

In Fig. 14, measured results for both the RS(544, 514, 15) KP4 and RS(528, 514, 7) KR4 codes are reported. Gray encoding is enabled to reduce BER. Different data points are generated by varying the amount of Gaussian-like crosstalk injected to the channel. To minimize the impact of random jitter, all data points are measured by locking the CDR phase and DFE tap weights once the DFE tap weights' LMS adaptation has converged. The curves generated by our statistical model are also superimposed in Fig. 14, treating the crosstalk as additive white Gaussian noise. Following the iterative procedure described in Section V-B, we select

[b] See Appendix for a justification.

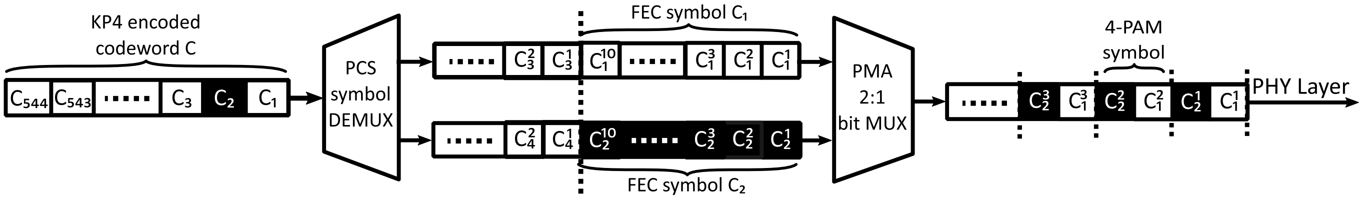

KP4 encoded
codeword C
C_{544} C_{543} ...... C_3 C_2 C_1
PCS
symbol
DEMUX
FEC symbol C_1
...... C_3^2 C_3^1 C_1^{10} ...... C_1^3 C_1^2 C_1^1
...... C_4^2 C_4^1 C_2^{10} ...... C_2^3 C_2^2 C_2^1
FEC symbol C_2
PMA
2:1
bit MUX
4-PAM
symbol
...... C_2^3 C_1^3 C_2^2 C_1^2 C_2^1 C_1^1
PHY Layer

Fig. 13. System-level diagram showing FEC symbol distribution and 2:1 bit multiplexing at TX.

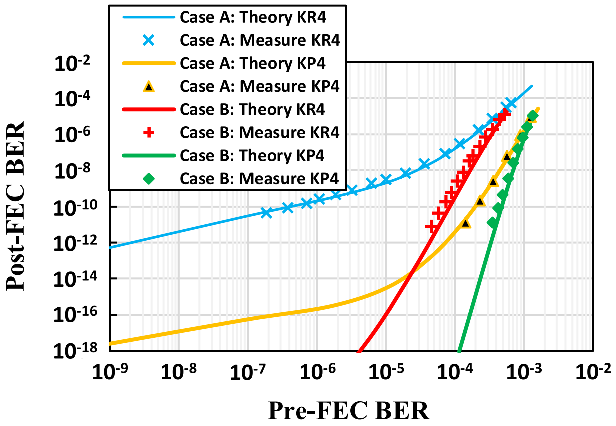


Fig. 14. Measured and theoretical pre-FEC vs post-FEC BER plot for RS(528, 514, 7) and RS(544, 514, 15) code.

$j_s^{max} = 20$ for the KP4 code and $j_s^{max} = 14$ for the KR4 code to ensure $e_r(j_s^{max}) \leq 2\%$ at a pre-FEC BER of 10^{-3} in both test cases.

All data points in Fig. 14 are measured down to a post-FEC BER of 10^{-11}. Good consistency is observed between the theoretical curves and measured results. The combined effect of many noise sources including ISI, crosstalk, and ADC quantization noise in wireline links has a pdf that is well-approximated by a Gaussian [5], [7]. Thus, the shape of the post-FEC vs pre-FEC BER curve is mainly dictated by the DFE taps weights. Moreover, for case A where a large amount of error propagation is present, our statistical model can properly predict the error floor with the RS(528, 514, 7) KR4 code. Importantly, our statistical model accurately predicts the measured transition between the two regions for the KR4 and KP4 FEC in case A. Furthermore, the model indicates that for the KP4 FEC, in order to ensure a post-FEC BER of 10^{-18}, a pre-FEC BER of 10^{-4} is adequate for case B, whereas a pre-FEC BER of 10^{-10} is required for case A, conclusions that would have been almost impossible to draw using the existing methods. Our statistical model can be used to quantify the precise pre-FEC BER required to achieve very low post-FEC BER depending on the channel and equalizer.

VII. Conclusion

This paper described a systematic and efficient method that can be used to accurately estimate post-FEC BER for high-speed wireline communication channels using standard linear block codes on GF(2^m). We proposed a two-level hierarchical statistical model allowing us to model the propagation of both PAM-symbol and FEC-symbol errors corrupting the FEC decoder. The model is simplified through a series of techniques including state aggregation, time aggregation, state reduction, and pruning-based trellis dynamic programming to accelerate the statistical analysis. The error bound associated with each method is also clearly defined. Because of the hierarchical approach, the time complexity of the analysis only depends on the FEC code but not the underlying PAM sub-trellises. An experimental prototype verified the proposed model where all measured results worked quite closely to that predicted by the theory.

TABLE I
Time Comparison for Statistical and Behavioral Models and 60 Gb/s Lab BERT using RS(544,514,15) Code

Post-FEC BER Level	Statistical Model *	Behavior Model †	60Gb/s BERT
$1\cdot10^{-8}$	~0.5min	200 hours	1.76 sec
$1\cdot10^{-12}$		1799 days	4.90 hours
$1\cdot10^{-15}$		4928 years	204 days

* with the more-complex RS(544,514,15) KP4 code
† accelerated by parallel processing

A comparison of simulation times using the statistical model, a behavioral Simulink model, and a laboratory 60 Gb/s bit error rate test (BERT) measurement are recorded in Table I. The statistical model has all simulation parameters identical to those reported in Fig. 14. The behavioral model is accelerated by parallel processing using a 16-core processor, resulting in 6.81 μs per bit in the simulation. The total time needed to simulate or measure three post-FEC BER levels are reported in the table, assuming each BER simulation or measurement must observe at least 1000 bit errors. Note that our statistical analysis results extend down to 10^{-15} or even further without increasing the number of calculations. At these low BER levels, the impact of error propagation is significant, but behavioral simulation and even laboratory BERT measurement are impractical. In addition, the statistical simulation can be prohibitively long without using the techniques introduced in this work to improve efficiency of the model. For example, according to (30) the statistical simulation performed in Table I would require a total number of 4.57×10^7 trellis node iterations assuming a KP4 code with $j_s^{max} = 20$. Without pruning, by (29) the FEC trellis model would instead require 1.08×10^{10} iterations, making the simulation time almost three orders of magnitude higher.

While this paper demonstrates the statistical analysis method in the presence of DFE error propagation, the method is general and can be applied to model other communication systems having memory effects. Moreover, our proposed model can be extended to higher-level PAM schemes and other advanced equalizer architectures to assist in making architectural choices for wireline transceivers such as co-design of the equalization and FEC in the presence of error propagation and various noise sources.

Appendix

According to (2) a single receiver error D_{k-1} results in an additive error at the receiver input

$$n_k^{dfe} = -D_{k-1}h_1. \tag{38}$$

If another error arises, the additive error at time $k + 1$ is

$$n_{k+1}^{dfe} = -D_{k-1}h_2 - D_k h_1. \tag{39}$$

If $h_1 > 0$ the sign of (38) is opposite that of the preceding error, thus increasing the probability of a new error D_k also having an opposing sign. In this case, since D_{k-1} and D_k have opposing signs, the two terms in (39) will add constructively resulting in the largest possible additive error term only if h_1 and h_2 have opposing signs, implying $h_2 < 0$. Alternatively, if $h_1 < 0$ the additive error (38) is of the same sign as D_{k-1} increasing the probability of another error D_k also having the same sign. In this case, the additive error (39) is increased when h_2 has the same sign as h_1; that is, when $h_2 < 0$. Thus, in either case the probability of propagating errors two or more time steps is maximized by a negative h_2.

To prove that the probability of having errors with the same sign is higher if $h_1 < 0$ and vice versa, we assume $D_{k-1} = \pm 2$ and an equal probability of transmitting $b_k \in \{\pm 1\}$. According to (3) the probability of $D_k = +2$ is

$$P_{+2} = \frac{1}{2} Q\left(\frac{-h_0 \mp 2h_1}{\sigma}\right). \tag{40}$$

Similarly, under the same assumption the probability of $D_k =$ -2 is

$$P_{-2} = \frac{1}{2} Q\left(\frac{-h_0 \pm 2h_1}{\sigma}\right). \tag{41}$$

With a positive h_0 and negative h_1, $P_{+2} > P_{-2}$ if $D_{k-1} = +2$ and $P_{-2} > P_{+2}$ if $D_{k-1} =$ -2. Therefore, it is much more likely that D_{k-1} and D_k have the same sign if $h_1 < 0$. Similarly, in (40) and (41) if $h_1 > 0$, it can be easily proven that D_{k-1} and D_k are likely to have opposing signs.

Acknowledgements

The authors would like to thank Professor Frank Kschischang from University of Toronto and Chris Feist from Huawei Canada for their expertise and assistance throughout this project.

References

[1] R. L. Narasimha and N. Shanbhag, "Forward error correction for high-speed I/O," in *Proc. 42nd Asilomar Conf. Signals, Syst. Comput.*, Pacific Grove, CA, USA, Oct. 2008, pp. 1513–1517.

[2] A. Alvarado, E. Agrell, D. Lavery, R. Maher, and P. Bayvel, "Replacing the soft-decision FEC limit paradigm in the design of optical communication systems," *J. Lightw. Technol.*, vol. 33, no. 20, pp. 4338–4352, Oct. 15, 2015.

[3] *IEEE Draft Standard for Ethernet Amendment 2: Physical Layer Specifications and Management Parameters for 100 Gb/s Operation Over Backplanes and Copper Cables*, IEEE Standard P802.3bj/D3.1, IEEE Std (802.3-2012), Apr. 2014, pp. 1–428.

[4] I. Oh and R. Harjani, "Adaptive techniques for joint optimization of XTC and DFE loop gain in high-speed I/O," *ETRI J.*, vol. 37, no. 5, pp. 906–916, Oct. 2015.

[5] S. Kiran *et al.*, "Modeling of ADC-based serial link receivers with embedded and digital equalization," *IEEE Trans. Compon. Packag. Manuf. Technol.*, vol. 9, no. 3, pp. 536–548, Mar. 2019.

[6] J. Kim *et al.*, "Equalizer design and performance trade-offs in ADC-based serial links," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 58, no. 9, pp. 2096–2107, Sep. 2011.

[7] V. Stojanovic and M. Horowitz, "Modeling and analysis of high-speed links," in *Proc. IEEE Custom Integr. Circuits Conf.*, Sep. 2003, pp. 589–594.

[8] K. S. Oh *et al.*, "Accurate system voltage and timing margin simulation in high-speed I/O system designs," *IEEE Trans. Adv. Packag.*, vol. 31, no. 4, pp. 722–730, Nov. 2008.

[9] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, no. 5, pp. 1253–1265, 1960.

[10] E. O. Elliott, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, no. 5, pp. 1977–1997, Sep. 1963.

[11] B. K. Casper, M. Haycock, and R. Mooney, "An accurate and efficient analysis method for multi-Gb/s chip-to-chip signaling schemes," in *Symp. VLSI Circuits Dig. Tech. Papers*, Honolulu, HI, USA, Jun. 2002, pp. 54–57.

[12] A. Szczepanek, I. Ganga, C. Liu, and M. Valliappan, *10GBASE-KR FEC Tutorial*. Accessed: May 1, 2019. [Online]. Available: http://www.ieee802.org

[13] *Transcoding/FEC Options and Trade-Offs for 100 Gb/s Backplane and Copper Cable*, IEEE Standard 802.3bj, Nov. 2011.

[14] *FEC Codes for 400 Gbps802.3bs*, IEEE Standard 802.3bs, Nov. 2014.

[15] K. Xiao, B. Lee, and X. Ye, "A Flexible and efficient bit error rate simulation method for high-speed differential link analysis using time-domain interpolation and superposition," in *Proc. IEEE Int. Symp. Electromagn. Compat.*, Detroit, MI, USA, Aug. 2008, pp. 1–6.

[16] R. Narasimha, N. Warke, and N. Shanbhag, "Impact of DFE error propagation on FEC-based high-speed I/O links," in *Proc. IEEE Global Telecommun. Conf.*, Honolulu, HI, USA, Nov./Dec. 2009, pp. 1–6.

[17] P. Monsen, "Adaptive equalization of the slow fading channel," *IEEE Trans. Commun.*, vol. COM-22, no. 8, pp. 1064–1075, Aug. 1974.

[18] V. Gaudet, "A survey and tutorial on contemporary aspects of multiple-valued logic and its application to microelectronic circuits," *IEEE J. Emerg. Sel. Topics Circuits Syst.*, vol. 6, no. 1, pp. 5–12, Mar. 2016.

[19] J. Kim *et al.*, "A 112Gb/s PAM-4 transmitter with 3-Tap FFE in 10nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2018, pp. 102–104.

[20] K. Gopalakrishnan *et al.*, "A 40/50/100Gb/s PAM-4 Ethernet transceiver in 28nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Jan./Feb. 2016, pp. 62–63.

[21] P. Upadhyaya *et al.*, "A fully adaptive 19-to-56Gb/s PAM-4 wireline transceiver with a configurable ADC in 16nm FinFET," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2018, pp. 108–110.

[22] L. Tang, W. Gai, L. Shi, X. Xiang, K. Sheng, and A. He, "A 32Gb/s 133mW PAM-4 transceiver with DFE based on adaptive clock phase and threshold voltage in 65nm CMOS," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2018, pp. 114–116.

[23] *FEC Performance on Multi-Part Links*, IEEE Standard 802.3bs, Nov. 2014.

[24] X. Dong, G. Zhang, and C. Huang, "Improved engineering analysis in FEC system gain for 56G PAM4 applications," in *Proc. DesignCon*, Santa Clara, CA, USA, 2018.

[25] C. Loi *et al.*, "6.5 A 400Gb/s transceiver for PAM-4 optical direct-detect application in 16nm FinFET," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2019, pp. 120–122.

[26] A. Leon-Garcia, *Probability, Statistics, and Random Processes for Electrical Engineering*. Upper Saddle River, NJ, USA: Prentice-Hall, 2007.

[27] R. Kennedy and B. Anderson, "Recovery times of decision feedback equalizers on noiseless channels," in *IEEE Trans. Commun.*, vol. 35, no. 10, pp. 1012–1021, Oct. 1987.

[28] J. G. Kemeny and J. L. Snell, *Finite Markov Chains*, 2nd ed. New York, NY, USA: Springer, 1976.

[29] G. Rubino and B. Sericola, "On weak lumpability in Markov chains," *J. Appl. Probab.*, vol. 26, no. 3, pp. 446–457, 1989.

[30] C. D. Meyer, "Stochastic complementation, uncoupling Markov chains, and the theory of nearly reducible systems," *SIAM Rev.*, vol. 31, no. 2, pp. 240–272, 1989.

[31] D. Bertsekas and J. Tsitsiklis, *Neuro-Dynamic Programming*. Nashua, NH, USA: Athena Scientific, Sep. 1996.

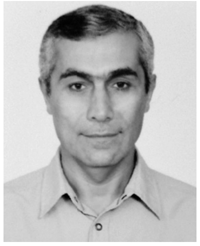
[32] X.-R. Cao, Z. Ren, S. Bhatnagar, M. Fu, and S. Marcus, "A time aggregation approach to Markov decision processes," *Automatica*, vol. 38, no. 6, pp. 929–943, Jun. 2002.

[33] R. McEliece and L. Swanson, "On the decoder error probability for Reed–Solomon codes (Corresp.)," *IEEE Trans. Inf. Theory*, vol. 32, no. 5, pp. 701–703, Sep. 1986.

[34] M.-A. Lacroix *et al.*, "6.2 A 60Gb/s PAM-4 ADC-DSP transceiver in 7nm CMOS with SNR-based adaptive power scaling achieving 6.9pJ/b at 32dB loss," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, San Francisco, CA, USA, Feb. 2019, pp. 114–116.

Ming Yang (S'14) received the B.Eng. degree in aerodynamic engineering from the Department of Aeronautics, Xiamen University, Xiamen, China, in 2012, and the B.Eng. and M.Eng. degree in electrical engineering from the Department of Electrical and Computer Engineering, McGill University, Montreal, Canada, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree with the Edward S. Rogers Sr. Department of Electrical & Computer Engineering, University of Toronto. His research interests are in analog integrated circuit design, on-chip analog signal processing, and high-performance integrated circuit testing.

Shayan Shahramian received the Ph.D. degree from the Department of Electrical and Computer Engineering, University of Toronto, Canada, in 2016. His focus is on high-speed chip-to-chip communication for wireline applications. He joined Huawei Canada, in January 2016, where he is currently working on system/circuit level design of high-speed energy-efficient transceivers for short reach applications. He is also a recipient of the NSERC Industrial Postgraduate scholarship in collaboration with Semtech Corporation (Gennum Products), the Best Young Scientist Paper Award at the ESSCIRC 2014, and the Analog Devices Outstanding Designer Award in 2014.

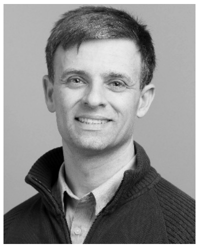


Hossein Shakiba received the B.Sc. and M.Sc. degrees in electrical engineering from the Department of Electrical and Computer Engineering, Isfahan University of Technology, Iran, in 1985 and 1989, respectively, and the Ph.D. degree in electrical engineering from the Department of Electrical and Computer Engineering, University of Toronto, Canada, in 1997. He has over 30 years of teaching, research, design, and management experience in the area of analog circuit and system design for various applications with a focus on wireline communication in the industry and academia. He is currently working on system and circuit design and development for next generation high-performance and high-efficiency serial links at Huawei Canada.

Henry Wong received the B.A.Sc. and Ph.D. degrees in electrical engineering. He has worked for Nortel, Cadence, and Lucent on high-speed modems, and Gennum (Semtech) on SerDes, CDR. He joined Huawei, in 2013, where he is also a Manager of SerDes system architecture product development. He is currently a Distinguished Engineer with Huawei. His area of R&D interest is in SerDes design, for high-speed interface, optical module, and backplane communications.

Peter Krotnev is currently a Sr. Principal Engineer with Huawei Technologies and also a member of the High Speed I/O System Development Team. He is also responsible for SerDes architecture improvements, electrical specifications, test planning, and leading the development of the SerDes tuning and adaptation strategies.

As a telecom professional, he has also worked with STMicroelectronics Inc. on a variety of projects and technologies, including ADSL, gigabit ethernet, and high-speed SerDes. As a signal integrity expert, he is also involved in number of patents and articles.



Anthony Chan Carusone (S'96–M'02–SM'08) received the Ph.D. degree from the University of Toronto in 2002. Since then, he has been a Professor with the Department of Electrical and Computer Engineering, University of Toronto. He is also an occasional consultant to industry in the areas of integrated circuit design and digital communication.

Prof. Chan Carusone coauthored the Best Student Papers at the 2007, 2008, and 2011 Custom Integrated Circuits Conferences, the Best Invited Paper at the 2010 Custom Integrated Circuits Conference, the Best Paper at the 2005 Compound Semiconductor Integrated Circuits Symposium, and the Best Young Scientist Paper at the 2014 European Solid-State Circuits Conference. He also coauthored, along with D. Johns and K. Martin, the second edition of the textbook *Analog Integrated Circuit Design*. He was the Editor-in-Chief of the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS in 2009 and an Associate Editor of the IEEE JOURNAL OF SOLID-STATE CIRCUITS from 2010 to 2017. He has served on the technical program committees for the Custom Integrated Circuits Conference and the VLSI Circuits Symposium. He was a Distinguished Lecturer of the IEEE Solid-State Circuits Society from 2015 to 2017. He currently serves as a member of the Technical Program Committee of the International Solid-State Circuits Conference.