On the Optimal Delay Amplification Factor of Multi-Hop Relay Channels

Dennis Ogbe, Chih-Chun Wang, and David J. Love
Purdue ECE, USA; Email: {dogbe,chihw,djl}@purdue.edu

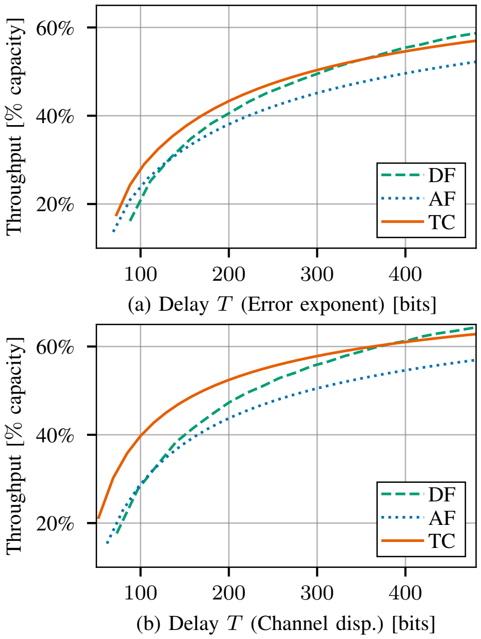
***Abstract*—The abstract model of the multi-hop relay channel is fundamental to a vast variety of modern communication systems. This fact, coupled with the demand for ultra-reliable-low-latency communication (URLLC), motivates a new investigation of relay channels from a delay-vs-throughput perspective. This work seeks to analyze this tradeoff in the regime of asymptotically large, yet still finite delay. A new metric called the *Delay Amplification Factor* (DAF) is introduced, which allows analytic comparison of the asymptotic delay across different relay solutions, e.g. decode-&-forward (DF), compress-&-forward, etc. The optimal DAF (over all possible existing/future designs) is then characterized for two special settings, one with fixed-length coding and one with variable-length coding and 1-bit stop feedback. The results show that under some general conditions, the optimal end-to-end delay over an L-hop line network is *asymptotically comparable* to the delay over the single bottleneck hop, and it does not grow linearly with respect to L. The linearly growing delay penalty commonly encountered in DF and other schemes is thus an artifact rather than a fundamental limit of multi-hop relay communication.**

I. Introduction

The relay channel [1] is a classic information theory problem which has experienced a renewed surge of interest due to its applicability to a vast variety of modern communication systems, including but not limited to traditional satellite relays, internet-of-things (IoT) network architectures, and self-backhauling techniques considered in 5G and beyond [2].

While the capacity of the general relay channel model remains an open problem, many practical scenarios, e.g., IoT and wireless self-backhauling [2], can be modeled as a simpler *separated relay channel*, which essentially concatenates two 1-hop channels while assuming that the destination cannot hear the source directly. See Section II for details. The capacity of this arguably more relevant model is $C = \min(C_1, C_2)$ and is achievable by the decode-&-forward (DF) policy [1]. While the capacity of the separated relay channel is well understood, with the recent focus on ultra-low-latency, e.g. sub-1ms in 5G URLLC [3], this work studies the delay of separated relay channels and we ask the following question: For a fixed throughput requirement, what is the transmission scheme that minimizes end-to-end delay? Broadly speaking, our work can be viewed as the multi-hop-relay counterpart for the finite-length analysis of the point-to-point channel in [4].

Existing finite-length analysis works [5]–[7] are based on well-known relay policies like DF, compress-&-forward, etc. While the schemes in [5]–[7] may perform well in a general relay channel model, none of them can outperform DF once we are limited to the separated relay channels, for which DF is already capacity achieving. To explore the optimal throughput-delay tradeoff (not limited by any existing designs), [8] devised a new scheme called *transcoding* (TC) that substantially outperforms both DF and amplify-&-forward (AF) in the finite blocklength regime. Fig. 1 plots the throughput-delay tradeoff of TC [8] using either Gallager's error exponent analysis or the channel dispersion results in [4]. The details of the computation of the curves were provided in [8].



(a) Delay T (Error exponent) [bits]

(b) Delay T (Channel disp.) [bits]

Fig. 1: Finite-blocklength results from [8] in which transcoding (TC) is presented. (a) per the error exponent analysis and (b) per the channel dispersion approximation.

The work in [8] provided a new construction and insight about how to improve the finite-length performance in the small-delay regime. However, a closer look at Fig. 1 shows that DF still outperforms TC when the delay is $\geq$ 350 symbol (bit) durations. To complement the small-delay results in [8], this paper aims to characterize the best possible throughput-vs-delay performance in the asymptotic long-delay regime. Key contributions of this paper are as listed as follows.

A new problem formulation. We consider an L-hop line network and introduce a new problem formulation and a new concept called *Delay Amplification Factor* (DAF) for any given

This work was supported in parts by NSF under Grant CCF-1422997, Grant ECCS-1407604, Grant CCF-1618475, and Grant CCF-1816013.

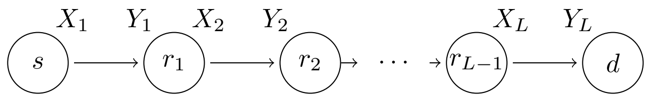


Fig. 2: L-hop line network

scheme. DAF quantifies the multiplicative increase of the end-to-end delay that arises when sending information over L hops as compared to the delay experienced by transmitting only over a single hop, more precisely, the *bottleneck* hop.

DAF analysis without feedback. We construct a scheme with DAF = 1, provided that the bottleneck hop is the last hop, i.e. the optimal end-to-end delay of an L-hop network is *asymptotically comparable* to the delay over its bottleneck hop in this scenario. The conventional belief that the delay of an L-hop relay channel grows linearly with respect to L is thus likely an artifact of the underlying DF policy.

DAF analysis with one-time stop feedback. It is known that 1-bit stop feedback significantly reduces the expected delay for point-to-point channels [9]. We show that similar delay benefits hold for multi-hop relays. Specifically, the aforementioned result is promising (DAF = 1) but is under a restrictive setting (bottleneck is the last hop). We prove that with the addition of one-time 1-bit stop feedback from the destination back to the source, optimal DAF = 1 is achievable *regardless of the location of the bottleneck.* That is, the number of hops in a line network has no fundamental multiplicative delay impact in the asymptotically-small-error-rate regime.

II. The Delay Amplification Factor of Multi-Hop Relay Channels

We consider the L-hop line network in Fig. 2 with slotted transmission for $t = 1, 2, \ldots$. One symbol is sent in each time slot. Each hop is discrete and memoryless and we denote the input and output symbols of the l-th hop at time slot t as $X_l(t)$ and $Y_l(t)$, respectively. Denote the capacity of the l-th hop by C_l (unit: nats/slot). For ease of exposition, we assume $C_l < \infty$ for all l and assume a unique bottleneck hop l^*, i.e. $\exists l^*$ such that $C_l > C_{l^*}, \forall l \neq l^*$. The overall capacity is then $C = C_{l^*}$ and we assume $C_{l^*} > 0$. Lastly, we denote the random coding error exponent [10] of hop l by[1] $E_{\mathsf{rc},l}(R)$.

Technical assumptions: We assume that the l-th hop channel distribution $p_l(y_l|x_l) > 0$ for all $l \in [1, L]$ and all input/output symbols x_l and y_l. This ensures that $E_{\mathsf{rc},l}(R)$ behaves properly without running into any corner cases. We can relax this assumption (e.g., to include binary erasure channels in our setting) by the conditions (a) and (b) in [10, p. 71].

Source s wishes to send an integer message M, drawn uniformly randomly from $\mathcal{M} = \{1, 2, \ldots, |\mathcal{M}|\}$ to destination d using the following transmission scheme.

Starting times and duration. A sequence of L deterministic, non-decreasing time points $\tau_1 = 0 \leq \tau_2 \leq \cdots \leq \tau_L$ determines the starting times (unit: slots) of data transmission for the corresponding hops. The total duration of the transmission at each node is denoted as T_{dur} (unit: slots). Fig. 3 gives an example for $T_{\text{dur}} = 9$ and starting times $\tau_1 = 0, \tau_2 = 2, \tau_3 = 3$, and $\tau_4 = 6$ over $L = 4$ hops.

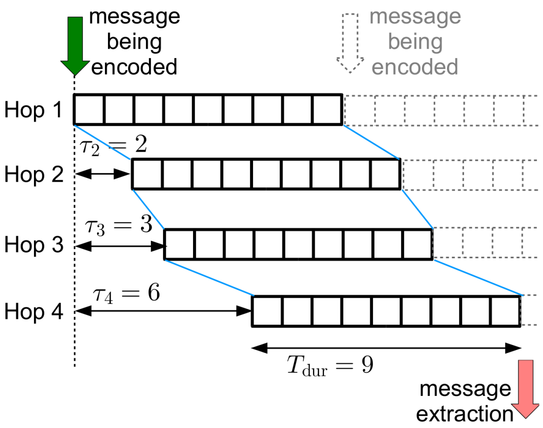


Fig. 3: Starting times, transmission duration, and encoding/decoding timepoints for an example scheme (unit: slots).

Sequential encoding at the relay nodes. We assume full-duplex relays with causal encoding. That is,

$$X_1(t) = f_t^{[1]}(M), \quad \forall t \in (\tau_1, \tau_1 + T_{\text{dur}}] \tag{1}$$

$$X_l(t) = f_t^{[l]}([Y_{l-1}]_*^{t-1}), \quad \forall l \geq 2,\ \forall t \in (\tau_l, \tau_l + T_{\text{dur}}] \tag{2}$$

where $f_t^{[l]}$ is the encoder of the l-th hop at time slot t, and

$$\begin{aligned}&[Y_{l-1}]_*^{t-1} \triangleq \\ &\quad \{Y_{l-1}(\tau) : \tau \in (\tau_{l-1}, \min(t-1, \tau_{l-1} + T_{\text{dur}})]\}\end{aligned} \tag{3}$$

denotes all previously received observations from the upstream hop.

Block decoding at the destination. The final block-based decoding function is given as

$$\widehat{M} = g([Y_L]_*^{\tau_L + T_{\text{dur}}}). \tag{4}$$

Definition 1: An L-hop line network scheme, described by the aforementioned three components, *attains* a delay-throughput-error-rate tuple (T, R, ϵ) if

$$T \geq \tau_L + T_{\text{dur}} \qquad \text{(unit: slots)} \tag{5}$$

$$R \leq \frac{\ln(|\mathcal{M}|)}{T_{\text{dur}}} \qquad \text{(unit: nats/slot)} \tag{6}$$

$$\epsilon \geq P\left(\widehat{M} \neq M\right). \tag{7}$$

For any given scheme Φ, we use $\mathcal{A}_\Phi$ as the collection of all tuples (T, R, ϵ) that can be attained by scheme Φ.

Definition 2: The error exponent of scheme Φ is defined as

$$E_\Phi(R) \triangleq \limsup_{T\to\infty} \sup_{\epsilon:(T,R,\epsilon)\in\mathcal{A}_\Phi} \frac{-\ln(\epsilon)}{T}. \tag{8}$$

A few remarks are in order at this stage.

Remark 1: When $L = 1$, our definition includes the traditional finite-length analysis [4] as a special case, since

[1] A more appropriate notation of the (optimal) random coding error exponent would be $E_{\mathsf{rc},l}(R, \mathbf{Q}^*(R))$, where $\mathbf{Q}^*(R)$ is the optimal prior distribution at the given rate R. However, for notational simplicity, we use $E_{\mathsf{rc},l}(R) \triangleq E_{\mathsf{rc},l}(R, \mathbf{Q}^*(R))$ as shorthand.

$\tau_1 = 0$ and since T_{dur} is simply the block length. For arbitrary L, it includes DF as a special case by setting $\tau_{l+1} = \tau_l + T_{\mathrm{dur}}$, i.e., each relay starts transmitting only after the upstream hop is finished.[2]

Remark 2: The throughput defined in (6) uses T_{dur} in the denominator rather than $(\tau_L + T_{\mathrm{dur}})$. This is because we allow the throughput to be enhanced by pipelining, which is illustrated by dotted lines/text in Fig. 3. One can see that the encoding of the next message starts before the rest of the network finishes the current message. Allowing for pipelining is crucial since it enables fair throughput assessment for schemes like DF and block Markov coding.

We are interested in the delay when operating at rates arbitrarily close to the capacity, i.e., $R \nearrow C_{l^*}$, and deem any not-capacity-achieving schemes as uninteresting. This thus leads to the following definition of the DAF.

Definition 3: The Delay-Amplification-Factor (DAF) of an L-hop communication scheme Φ is defined as

$$\Gamma_\Phi \triangleq \lim_{R \nearrow C_{l^*}} \frac{E_{\mathsf{rc},l^*}(R)}{E_\Phi(R)}. \tag{9}$$

The rationale behind is that in the asymptotic regime (a fixed but infinitesimal ϵ) the smaller the error exponent is, the longer the delay needed to attain the target ϵ. Therefore, $E_{\mathsf{rc},l^*}(R)$ is generally larger than $E_\Phi(R)$ since the former focuses on the delay over the bottleneck hop and the latter focuses on the longer end-to-end delay. The ratio thus signifies the multiplicative delay impact of scheme Φ when compared to the delay experienced by the bottleneck hop. We then have

Lemma 1: Regardless of the scheme Φ, we always have

$$\Gamma_\Phi \geq 1. \tag{10}$$

Lemma 2: For decode-&-forward (DF) schemes, we have

$$\Gamma_{\mathrm{DF}} = \sum_{l=1}^{L} \frac{C_{l^*}}{C_l}. \tag{11}$$

The intuition behind these lemmas is as follows. Since the optimal delay of an L-hop line network is no better than the optimal delay over its bottleneck hop, DAF is lower bounded by 1 due to the sphere-packing bound results. Using DF to send a B-nats message over all L hops takes roughly $\sum_l \frac{B}{C_l}$ symbols and it takes only $\frac{B}{C_{l^*}}$ symbols to send through the bottleneck hop. Being the ratio of the delays, the DAF of DF becomes (11). The proofs are omitted due to space limits.

Comparison to existing results: [7] characterizes the error exponent of $(L = 2)$-hop relays under DF, partial DF (PDF), and compress-&-forward (Comp-F). Block Markov Coding (BMC) is analyzed where $b \geq 2$ is the number of blocks. The parameter b decides the effective throughput $R_{\mathrm{eff}} = \frac{b-1}{b} R$ and in our setting we can always assume $R_{\mathrm{eff}} = R$ since we recover any effective throughput reduction of BMC by pipelining. Then a close look at [7] shows that the error exponents of DF, PDF, and Comp-F all exhibit a reduction factor $\frac{1}{b}$ (thus at least 1/2). Plugging in the DAF definition (9), the corresponding DAF ≥ 2, which is consistent with (11) (assuming $C_1 = C_2$) since the delays of DF, PDF, and Comp-F, all grow linearly with respect to L. ($L = 2$ in this discussion.)

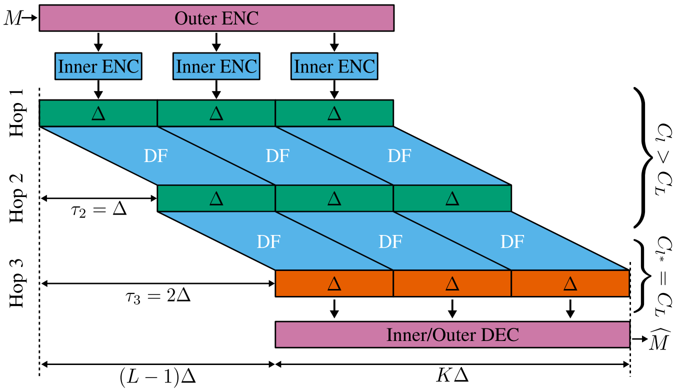


Fig. 4: Inner/outer code transmission scheme. (Example drawn for $L = 3$ and $K = 3$)

III. Main Results

A. The Optimal DAF without Feedback

Proposition 1 (Optimal DAF): If $l^* = L$, then we can construct a transcoding scheme Φ_{tc} such that $\Gamma_{\Phi_{\mathsf{tc}}} = 1$.

Proposition 2: For the case of $l^* \neq L$, define $l_0^* = 0$ and iteratively define $l_i^* = \arg\min_{l \in (l_{i-1}^*, L]} C_l$ for $i = 1, 2, 3, \cdots$ until $l_i^* = L$. Suppose there are I such l_i^*, i.e., $l_I^* = L$. Also assume the minimum is unique when computing each l_i^*. Then we can construct a scheme Φ such that

$$\Gamma_\Phi = \sum_{i=1}^{I} \frac{C_{l^*}}{C_{l_i^*}}. \tag{12}$$

We can prove that (12) is strictly less than (11) except for the case of $C_1 < C_2 < \cdots < C_L$. That is, if $l^* \neq L$, even though we do not know how to attain the lower bound DAF ≥ 1, we can design a scheme with strictly smaller DAF than the DF scheme (i.e., having strictly shorter asymptotic delay) in all but the special case of $C_1 < C_2 < \cdots < C_L$. Proposition 2 can be proved by combining the DF principle and the scheme Φ_{tc} in Proposition 1. We omit the details due to space limits.

We now sketch the proof of Proposition 1. Our scheme is inspired by the transcoding design [8] and the concatenated coding structure [11], [12]. In the sequel, we assume $l^* = L$, use Fig. 4 for illustration, and use the term *micro-block* when referring to the "inner block code" in our design.

Choosing the parameter K. For any $R < C_{l^*} = C_L$, we choose the largest integer $K \geq 1$ simultaneously satisfying

$$K \cdot E_{\mathsf{rc},L}(R) < \min_{l \in [1, L-1]} E_{\mathsf{rc},l}(C_L) \tag{13}$$

$$K \cdot (C_L - R) \leq C_L. \tag{14}$$

Note that this choice of K is always possible when we start with an R that is sufficiently close to C_L. Furthermore, when

[2] This high-level discussion assumes $C_1 = C_2 = \cdots = C_L$. If not, the proposed framework can still fully represent DF but the values of τ_l and T_{dur} need to be carefully chosen since hops with larger C_l will finish their transmission sooner and thus do not need to use up the entire T_{dur}.

$R \nearrow C_L$, we have $K \to \infty$. In the following discussion, we assume a fixed $R < C_L$ and thus a fixed K.

Operation at the source s. Consider K micro-blocks, each containing Δ symbols. For each micro-block, we construct a random codebook of rate R_I using the capacity-achieving marginal distribution $P^*_{X_1}$ for hop-1. We set the value $R_I = C_L$. Each codebook has $e^{\Delta R_I}$ codewords and is constructed independently randomly from other random codebooks. Fix the K codebooks and denote the i-th codeword of the k-th codebook by $\mathbf{c}_i^{[k]}$, a length-Δ row vector, where $i = 0, \cdots, e^{\Delta R_I} - 1$. We now define a *cyclically shifted outer code* (CSOC) as a collection of length-$(K\Delta)$ row vectors:[3]

$$\mathcal{C}_O = \left\{ (\mathbf{c}_{i_1}^{[1]}, \cdots, \mathbf{c}_{i_K}^{[K]}) : \left(\sum_{k=1}^{K} i_k \bmod e^{K\Delta(R_I - R)} \right) = 0 \right\} \tag{15}$$

Namely, each codeword of $\mathcal{C}_O$ is a concatenation of K codewords, $\mathbf{c}_{i_k}^{[k]}, \forall k = 1 \cdots, K$, each from a different inner codebook, such that the sum of the subscripts $\sum_{k=1}^{K} i_k$ is a multiple of $e^{K\Delta(R_I - R)}$.

We then note that in general, a modulo-x operation "slices" the overall space into x equal-sized partitions. Since the total space of $(i_1, \cdots, i_K)$ is of size $e^{K\Delta R_I}$, the modulo-$\left(e^{K\Delta(R_I-R)}\right)$ operation thus ensures that the codebook $\mathcal{C}_O$ has $e^{K\Delta R_I}/e^{K\Delta(R_I-R)} = e^{K\Delta R}$ codewords, provided we assume some very mild divisibility condition.

Since there are $e^{K\Delta R}$ codewords, the source s uses $\mathcal{C}_O$ to send a message $M \in [1, e^{K\Delta R}]$ over the first hop, which takes $T_{\text{dur}} = K\Delta$ time slots to finish. The description of the source encoding function $f_t^{[1]}$ in (1) is complete.

Operation of the l-th relay with $l \geq 2$. Decode-&-forward (DF) is performed at hops 2 to L on a mico-block basis using the inner codebooks. Specifically, at the beginning of the k-th micro block (time $t = k\Delta + 1$), each relay will use the random inner codebook of its upstream hop to DECODE the index $i_{k-1} \in \{1, 2, \cdots, e^{\Delta R_I}\}$ transmitted by its upstream node in the previous micro-block (the $(k-1)$-th).

Then it will generate its own random inner codebook of rate R_I using its own capacity-achieving marginal distribution $P^*_{X_l}$. In total there are $e^{\Delta R_I}$ codewords in this codebook and we denote the i-th codeword by $\mathbf{c}_i^{[k]}$, which is a length-Δ row vector for all $i = 0, \cdots, e^{\Delta R_I} - 1$. The superscript k signifies that this random codebook, generated by hop l, is used only for the k-th micro-block. For the next micro-block, a completely new random inner codebook will be generated by hop l. After the creation of the inner codebook, the l-th relay FORWARDS the decoded i_{k-1} by sending $\mathbf{c}_{i_{k-1}}^{[k]}$. See Fig. 4 for illustration.

Operation at the destination d. At the end of time $t = ((L-1)+K)\Delta$, all the inner micro-blocks have been sent from s to d by DF, see Fig. 4. Destination d then performs optimal maximum likelihood (ML) decoding assuming the knowledge of all channel statistics $p_l(y_l|x_l)$, all the $K \cdot L$ inner random codebooks, and the CSOC used at source s.

The rest of the proof is to analyze the performance of the above scheme. The main argument hinges on the fact that since the K value satisfies (13), the error rate of DF during hops 1 to $(L-1)$ is negligible when compared to the errors in the last hop once we let $\Delta \to \infty$. Fig. 4 illustrates this with the difference in color between the green micro-blocks on the first $L-1$ hops and the red micro-blocks on the bottleneck hop. As a result, the dominant error event can be analyzed by assuming all first $(L-1)$ hops are error-free and by carefully characterizing the joint effects of the CSOC outer code and the random inner codes at the L-th hop. The detailed analysis is similar to the results in [13].

Letting $\Delta \to \infty$, we have the error exponent being

$$E_{\mathsf{rc},\Phi_{\mathsf{tc}}}(R) = \frac{K}{K+L-1} E_{\mathsf{rc},L}(R). \tag{16}$$

The final step is to notice that when $R \nearrow C_L$ the corresponding $K \to \infty$ simultaneously, which completes the proof of Proposition 1.

Remark 3: Concatenated coding for line networks was also used in [12] under a half-duplex, Gaussian channel setting. Nonetheless, a suboptimal two-stage *hard decoding* (decode the inner codes first, and then use the hard decisions to decode the outer code) was used in [11], [12], which significantly decreases the error exponent (see [11, Eq. (101)]) and is thus strictly suboptimal. In contrast, we prove that with our CSOC design, joint ML decoding at d attains the optimal DAF = 1.

B. The Optimal DAF with one-time stop feedback

We now change our formulation to allow for one-time one-bit stop feedback. For this, we fix $\tau_1 = \tau_2 = \cdots = \tau_L = 0$ and allow T_{dur} to be a stopping time of the filtration generated by $[Y_L]_1^t$. The encoder and decoder definitions from (1)–(4) remain identical. We replace the delay and throughput requirements in (5) and (6) by

$$T \geq \mathbb{E}\left[T_{\text{dur}}\right] \qquad \text{(unit: slots)} \tag{17}$$

$$R \leq \frac{\ln(|\mathcal{M}|)}{\mathbb{E}\left[T_{\text{dur}}\right]} \qquad \text{(unit: nats/slot).} \tag{18}$$

The error probability definition (7) and the error exponent definition (8) remain the same.

The above mathematical formulation models the following operations. The destination d can decide when to stop the whole "session" and start decoding the message $\widehat{M}$. Once the stop decision is made, a *1-bit stop feedback* message [9] is sent from d back to s, which propagates backwards through the line network. The entire network, including the source, the $L-1$ relays, and the destination, will then switch to the transmission of the next message. This variable-length setup is closely related to the concepts of hybrid-ARQ and digital fountain codes. Note that no feedback of any other form, e.g., per-symbol-feedback, per-hop feedback, relay-initiated feedback, etc., is ever allowed. The justification is that feedback is very costly from a delay's perspective. Arguably, the only

[3] Technically, an outer codebook [11] should be a collection of $(i_1, i_2, \cdots, i_K)$ and the length-$K\Delta$ vector $(\mathbf{c}_{i_1}^{[1]}, \cdots, \mathbf{c}_{i_K}^{[K]})$ is the end result after concatenating the outer and the inner codes. For ease of exposition, we abuse the notation and define $\mathcal{C}_O$ as in (15).

feasible option is the simple one-time, end-to-end, one-bit stop feedback.

Results in [9] show that for a point-to-point channel, stop feedback improves the random-coding error exponent from $E_{\mathsf{rc},l}(R)$ to a strictly larger value

$$E_{\mathsf{sf},l}(R) = (C_l - R)^+ \tag{19}$$

which we will use in the following new definition of the DAF for variable-length coding.

Definition 4: The DAF on an L-hop variable-length stop-feedback scheme Φ is defined as

$$\Gamma_\Phi \triangleq \lim_{R \nearrow C_{l^*}} \frac{E_{\mathsf{sf},l^*}(R)}{E_\Phi(R)} = \lim_{R \nearrow C_{l^*}} \frac{C_{l^*} - R}{E_\Phi(R)}. \tag{20}$$

Our second main result can then be stated as follows.

Proposition 3: We can construct a stop-feedback transcoding scheme Φ_{sftc} such that $\Gamma_{\Phi_{\mathsf{sftc}}} = 1$ regardless of whether $l^* = L$ or not.

For ease of exposition and due to space constraints, we describe some key components of Φ_{sftc} without detailed scheme descriptions.

Component 1: Concatenated coding with Sequential Random Permutation Outer Codes (SRPOC). Random inner codes are used for each micro block in the same way as in the fixed-length scheme Φ_{tc}. But Φ_{sftc} replaces the fixed-length CSOC outer code with a SRPOC outer code that continuously picks new index i_k for the k-th micro block and transmits codeword $\mathbf{c}_{i_k}^{[k]}$ from the k-th inner codebook.

Component 2: Sequential probability ratio test (SPRT). In part of our design, we apply the well-known technique of SPRT to the (sequential) concatenated coding with SRPOC and random inner codes. Some related results can be found in [9], [14], which analyzed SPRT but based on a non-concatenated random code construction.

Component 3: Micro-block-based DF. Similar to Φ_{tc}, we perform micro-block-based DF in Φ_{sftc}.

Component 4: The correction phase. We notice that micro-block-based DF operates on a smaller block length Δ and is thus subject to higher error rate. As a result, the relay node, in particular the receiver of the bottleneck hop, would continuously use SPRT and SRPOC+inner-code to decode the message with lower error rate, in parallel to the inner-code-based DF. After obtaining the higher-fidelity decision $\widehat{M}$ (based on SPRT and SRPOC), the relay will compare it to the past inner-code-based DF decisions. New "correcting signals" will then be sent to rectify the erroneous DF decisions made in the past. Note that this correction phase does not require any feedback and the relay(s) simply run SPRT+SRPOC in parallel with the inner-block DF and correct past erroneous decisions on a need basis.

Component 5: Carefully generated control messages. The overall scheme Φ_{sftc} is highly non-linear and requires passing crucial control information from the upstream to the downstream nodes, e.g., when to start the correction phase if the need arises, in a way similar to running a network protocol over the line network. Since there is no separate control channel, any control messages must be carried in the forward data channel. As a result, the format of the control messages needs to be carefully designed in order to manage the control overhead in terms of both throughput and delay.

IV. Conclusion & Future Work

We introduced and analyzed the Delay Amplification Factor (DAF) for L-hop line networks in this paper, which is motivated by our investigation into the best possible throughput-vs-delay relay performance started in [8]. While the work in [8] covered the small-delay regime, this text focused on the asymptotic regime and designed DAF-optimal coding schemes for some scenarios of the L-hop line networks, see Table I. A future direction is to characterize the optimal feedback-free DAF when the bottleneck hop is not the last, i.e., $l^* \neq L$.

	$l^* = L$	$l^* \neq L$
w/o feedback	This work: DAF = 1	Optimal DAF remains open
w/ stop feedback	This work: DAF = 1	This work: DAF = 1

TABLE I: Problem space breakdown for asymptotic analysis of delay/throughput tradeoff for L-hop line networks. l^* is the bottleneck hop.

References

[1] T. Cover and A. E. Gamal, "Capacity theorems for the relay channel," *IEEE Trans. Inform. Theory*, vol. 25, no. 5, pp. 572–584, September 1979.

[2] R.-A. Pitaval, O. Tirkkonen, R. Wichman, K. Pajukoski, E. Lahetkangas, and E. Tiirola, "Full-duplex self-backhauling for small-cell 5g networks," *IEEE Wireless Commun.*, vol. 22, no. 5, pp. 83–89, 2015.

[3] H. Ji, S. Park, J. Yeo, Y. Kim, J. Lee, and B. Shim, "Ultra-reliable and low-latency communications in 5g downlink: Physical layer aspects," *IEEE Wireless Commun.*, vol. 25, no. 3, pp. 124–130, JUNE 2018.

[4] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Channel coding rate in the finite blocklength regime," *IEEE Trans. Inform. Theory*, vol. 56, no. 5, pp. 2307–2359, May 2010.

[5] S. L. Fong and V. Y. F. Tan, "Achievable rates for gaussian degraded relay channels with non-vanishing error probabilities," *IEEE Trans. Inform. Theory*, vol. 63, no. 7, pp. 4183–4201, July 2017.

[6] Y. Hu, J. Gross, and A. Schmeink, "On the capacity of relaying with finite blocklength," *IEEE Trans. Veh. Technol.*, vol. 65, no. 3, pp. 1790–1794, March 2016.

[7] V. Y. F. Tan, "On the reliability function of the discrete memoryless relay channel," *IEEE Trans. Inform. Theory*, vol. 61, no. 4, pp. 1550–1573, April 2015.

[8] C.-C. Wang, D. J. Love, and D. Ogbe, "Transcoding: A new strategy for relay channels," in *55th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, Oct 2017, pp. 450–454.

[9] Y. Polyanskiy, H. V. Poor, and S. Verdu, "Feedback in the non-asymptotic regime," *IEEE Trans. Inform. Theory*, vol. 57, no. 8, pp. 4903–4925, Aug 2011.

[10] C. Shannon, R. Gallager, and E. Berlekamp, "Lower bounds to error probability for coding on discrete memoryless channels. ii," *Information and Control*, vol. 10, no. 5, pp. 522–552, 1967.

[11] G. D. Forney, *Concatenated codes.* Cambridge, MA, USA: MIT Press, 1966.

[12] N. Wen and R. A. Berry, "Reliability constrained packet-sizing for linear multi-hop wireless networks," in *2008 IEEE International Symposium on Information Theory*, July 2008, pp. 16–20.

[13] C. Thommesen, "Error-correcting capabilities of concatenated codes with MDS outer codes on memoryless channels with maximum-likelihood decoding," *IEEE Trans. Inform. Theory*, vol. 33, no. 5, pp. 632–640, Sep. 1987.

[14] M. V. Burnashev, "Data transmission over a discrete channel with feedback. random transmission time," *Problems Inform. Transmission*, vol. 12, no. 4, pp. 10–30, 1976.