Adaptive Doping of Spatially Coupled LDPC codes

Min Zhu*, David G. M. Mitchell†, Michael Lentmaier‡, and Daniel J. Costello, Jr.§
*State Key Laboratory of ISN, Xidian University, Xi'an, P. R. China, zhunanzhumin@gmail.com
†Klipsch School of Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM, USA, dgmm@nmsu.edu
‡Department of Electrical and Information Technology, Lund University, Lund, Sweden, michael.lentmaier@eit.lth.se
§Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN, USA, dcostel1@nd.edu

Abstract—**In this paper, we study the problem of error propagation in sliding window decoding (SWD) of spatially coupled LDPC (SC-LDPC) codes. A general decoder model that accounts for error propagation is proposed and analyzed, and the decoded block error rate (BLER) is calculated using the model. In order to improve the BLER performance under decoder error propagation conditions, adaptive variable node (VN) doping is proposed, assuming a noiseless binary feedback channel is available. Example calculations using the proposed model, as well as numerical simulation results, are used to show that adaptive VN doping improves the BLER performance compared to the periodic VN doping and to the undoped case.**

Index Terms—**Spatially Coupled Codes, Sliding Window Decoding, Code Doping**

I. Introduction

Spatially coupled low-density parity-check (SC-LDPC) codes, first introduced in [1], have drawn considerable attention from the research community in recent years due mainly to two facts: (1) with low-complexity iterative belief propagation (BP) decoding they can achieve the *maximum a-posteriori (MAP)* decoding threshold of an underlying *LDPC block code (LDPC-BC)*, termed the *threshold saturation effect* [2], [3], and (2) they can be decoded using *sliding window decoding (SWD)* [4], which greatly reduces decoding latency and complexity compared to other decoding methods. In order to maintain near optimal performance at moderate-to-high *signal-to-noise ratios (SNRs)*, the *window size* W should satisfy $W \geq 6v$, where v is the *decoding constraint length*. However, when low latency operation is desired at the lower SNRs typically used in applications, thus requiring smaller values of W, infrequent but severe decoder error propagation can sometimes result, causing significant performance degradation, particularly for large frame lengths. Also, in a continuous (streaming) transmission scenario, an unterminated code chain can result in unlimited decoder error propagation.

During the decoding process, error propagation is triggered when, after a block decoding error occurs, the decoding of the subsequent block is also affected, which in turn can cause a continuous string of block errors, resulting in an unacceptable performance loss. To address this problem, Klaiber et al. [5] proposed adapting the number of decoder iterations and/or shifting the window position in order to limit the effects of error propagation in SWD of SC-LDPC codes. Also in [6], we proposed a window extension algorithm, a synchronization mechanism, and a retransmission strategy to mitigate error propagation in SWD of *braided convolutional codes (BCCs)*, a type of SC-LDPC code. Each of these approaches was directed at altering the design of the *decoder*. More recently, we proposed *check node (CN) doped SC-LDPC codes* [7] and *variable node (VN) doped SC-LDPC codes* [8] to limit error propagation by altering the *encoder* design. This was accomplished by inserting doping points periodically into the code chain to help the decoder recover from error propagation. However, the required pre-determined distribution of doping points does not completely eliminate error propagation, since it can still exist between doping points. In order to address this deficiency, more doping points are required, but this results in a significant rate loss.

In this paper, we propose an adaptive doping strategy for SC-LDPC codes. Instead of inserting doping points periodically in a pre-determined way, we insert doping points only as needed, based on the average *log-likelihood ratios (LLRs)* in some number of recently decoded blocks. We refer to this strategy, which assumes the use of an instantaneous noiseless feedback channel, as *adaptive code doping*. Compared to periodic doping, it has the advantage of an almost immediate truncation of error propagation events while limiting the rate loss to only what is required to stop the error propagation. The adaptive approach can be applied to both CN doping and VN doping. Since both methods have similar performance, but VN doping is easier to implement, we focus here only on adaptive VN doping.

II. Modeling The Behavior of SWD

In this section, we introduce a new decoder model to illustrate how error propagation affects the *block error rate (BLER)* performance of SWD. In [6] and [7], a simple model of SWD behavior during error propagation was given. The model was focused on the case of a large *protograph lifting factor* M, corresponding to a strong code that typically performs very well until error propagation begins, but never recovers from this condition. In this section, we propose an extended decoder model, in which we allow for the possibility that the decoder can recover from a burst of decoded errors and resume correct decoding. This situation typically occurs for weaker codes with smaller values of M, where the use of past incorrectly decoded symbols in the decoding window does not have the same influence on future decoded blocks as it does for larger values of M.

Before proceeding with the development of the new decoder model, we first briefly review protograph-based SC-LDPC codes. We consider SC-LDPC codes constructed by coupling together a sequence of L disjoint (J, K)-*regular* LDPC-BC protographs into a single coupled chain, where infinite L results in an *unterminated* coupled chain and finite L results in a *terminated* coupled chain. Without loss of generality, we consider (3,6)-regular SC-LDPC codes constructed from

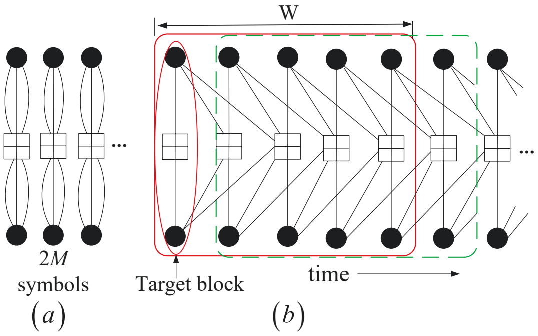

Fig. 1. (a) a sequence of independent (uncoupled) protographs; (b) spreading edges to the $m = 2$ nearest neighbors.

protographs as shown in Fig. 1. As an example, we begin with an independent (uncoupled) sequence of (3,6)-regular LDPC-BC protographs with *base matrix* $\mathbf{B} = [3 \;\; 3]$ (see Fig. 1(a)). The unterminated (3,6)-regular SC-LDPC code chain is obtained by applying the *edge-spreading* technique of [9] to the uncoupled protographs. In this case, the edge spreading is defined by a set of component base matrices $\mathbf{B}_0 = \mathbf{B}_1 = \mathbf{B}_2 = [1 \;\; 1]$ that satisfy $\mathbf{B} = \mathbf{B}_0 + \mathbf{B}_1 + \mathbf{B}_2$ (see Fig. 1(b)). In general, an arbitrary edge spreading must satisfy $\mathbf{B} = \sum_{i=0}^{m} \mathbf{B}_i$, where m is referred to as the *coupling width*. Applying the lifting factor M to the SC-LDPC protograph of Fig. 1(b) results in an unterminated ensemble of (3,6)-regular SC-LDPC codes in which each time unit represents a *block* of $2M$ coded bits (variable nodes). SWD, first proposed in [4], was applied to SC-LDPC codes to reduce decoding latency, memory, and complexity. As shown in Fig. 1(b), the rectangular box represents a decoding window of size W blocks. To decode, (1) a BP flooding schedule is applied to all the nodes in the window for some fixed number of iterations, or until some stopping criterion is met, (2) the *target block* of $2M$ symbols in the first window position is decoded according to the signs of their LLRs, and (3) the window shifts one time unit (block) to the right. Decoding continues in the same fashion until the entire chain is decoded, where the decoding latency in bits is given by $2MW$.

In order to obtain an understanding of decoder error propagation in SWD, we assume the decoder operates in one of three states: (1) a *random error state* S_{re0}, in which, given that the previous block was decoded correctly, decoding errors in the next block occur independently with probability p_0, resulting in the decoder transitioning to state S_{re1}, and the decoder remains in state S_{re0} following a correctly decoded block with probability $1 - p_0$, (2) a *random error state* S_{re1}, in which, given that the previous decoded block was in error, decoding errors in the next block occur independently with probability p_1, resulting in the decoder transitioning to state S_{be}, and the decoder returns to state S_{re0} following a correctly decoded block with probability $1-p_1$, and (3) a *burst error state* S_{be}, in which, given that the previous two decoded blocks were in error, decoding errors in the next block occur independently with probability $1-r$, resulting in the decoder staying in state S_{be}, and the decoder returns to state S_{re0} following a correctly decoded block with probability r.

Fig. 2 shows the state diagram describing this situation,

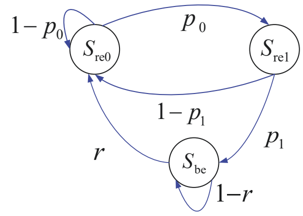

Fig. 2. The state diagram describing a decoder subject to error propagation.

where we assume the decoder always starts in state S_{re0}. Generally, $p_1 \geq p_0$ since one error block means that there are some incorrect LLRs still connected to the window that influence the decoding of the next block. Here, however, for simplicity we assume $p_1 = p_0 = p$. With this assumption, the probability of transitioning from state S_{re0} to S_{re1} is p and the probability of transitioning from state S_{re1} to S_{be} is also p, i.e., the probability of transitioning from state S_{re0} to state S_{be} is p^2. When $r \to 0$, and hence the error probability in state S_{be} is $1 - r \to 1$, error propagation occurs, i.e., the decoder will typically not be able to escape the burst error state. On the other hand, when $r > 0$, the decoder has probability r of returning to state S_{re0}, and the burst errors are of finite length.

For a given protograph, the *channel parameter SNR*, the *decoder parameter* W, and the *code parameter* M will all influence the values of p and r. p is a non-increasing function of all three of these parameters, whereas r increases with SNR and W, but decreases with M, since the larger number of incorrectly decoded LLRs still connected to the window will have a stronger influence on future decoded blocks in this case. In other words, stronger codes (large M) are less likely to reach state S_{be}, but once there, they have a higher probability of staying there, i.e., a higher probability of unlimited decoder error propagation. Weaker codes (small M), on the other hand, will have larger values of p, and thus will enter into state S_{be} more often, but are less likely to suffer from unlimited decoder error propagation. Instead, they typically result in more single errors and a larger number of burst errors of varying lengths. In the following, we will derive expressions for the BLER, as functions of p and r, of SC-LDPC codes based on this model for both unterminated ($L \to \infty$) and terminated (finite L) transmission.

A. *Asymptotic ($L \to \infty$) Analysis*

Let P_0 be the probability of being in state S_{re0}, P_1 be the probability of being in state S_{re1}, and P_{be} be the probability of being in state S_{be}. Then we have

$$P_0 = P_1\left(1-p\right) + P_{\mathrm{b}} r + P_0\left(1-p\right) = \left[P_1\left(1-p\right) + P_{\mathrm{b}} r\right]/p,$$
$$P_1 = P_0 p,$$
$$P_{\mathrm{be}} = P_1 p + P_{\mathrm{b}}\left(1-r\right) = P_0 p^2 + P_{\mathrm{b}}\left(1-r\right) = P_0 p^2/r.$$

The average BLER can now be expressed as

$$P_{\mathrm{BL}} = P_0 p + P_1 p + P_{\mathrm{be}}\left(1-r\right) = P_0\left(\frac{rp+p^2}{r}\right). \qquad (1)$$

Since $P_0 + P_1 + P_{\mathrm{b}} = \left(1 + p + p^2/r\right) P_0 = 1$, we have

$$P_0 = \frac{1}{\left(1 + p + p^2/r\right)} = \frac{r}{r + pr + p^2}. \qquad (2)$$

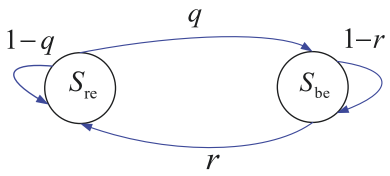

Fig. 3. The state diagram describing of a decoder subject to error propagation for finite L.

Now, using (2) in (1), we obtain

$$P_{\mathrm{BL}} = \frac{rp + p^2}{r + rp + p^2}, \tag{3}$$

where $P_{\mathrm{BL}} = 1$ when $r = 0$, which corresponds to the decoder entering state S_{be} once and never returning to state S_{re0} (the error propagation case); $P_{\mathrm{BL}} = \frac{p+p^2}{1+p+p^2}$ when $r = 1$, which corresponds to the decoder never making more than two consecutive errors (due to an error propagation mitigation method, say); and $P_{\mathrm{BL}} = p$ when $r = 1 - p$, which corresponds to a pure random error decoder. Since $\frac{\partial P_{\mathrm{BL}}}{\partial r} = -\frac{p^2}{(r+rp+p^2)} < 0$ for all $p > 0$, we see that P_{BL} decreases with r for all p, $0 < p \leq 1/2$.

B. Finite L Analysis

For a frame of length L, we group states S_{re0} and S_{re1} in Fig. 2 into a single random error state S_{re}, as shown in Fig. 3, where the transition probability from state S_{re} to state S_{be} is $q \approx 2p^2$.[1] Based on this model, the decoder stays in (the combined random error) state S_{re} for an average of $1/q$ time units and in state S_{be} for an average of $1/r$ time units. Thus, a "round trip" from S_{re} to S_{be} and back to S_{re} takes an average of $x = 1/q + 1/r$ time units, and the average number of round trips in a frame of length L is $y = L/x$. Now write $y = \lfloor y \rfloor + z$, where $z < 1$, and let $u = r/(r+q) < 1$ be the (average) fraction of a round trip that the decoder spends in state S_{re} and $v = q/(r+q) = 1 - u < 1$ be the (average) fraction of a round trip that the decoder spends in state S_{be}.

Now let n_{re} be the average number of time units spent in state S_{re} and let n_{be} be the average number of time units spent in state S_{be}. Then,

$$\begin{aligned} n_{\mathrm{re}} &= (\lfloor y \rfloor + z/u)\,(1/q)\,, n_{\mathrm{be}} = \lfloor y \rfloor\,(1/r), z < u, \\ n_{\mathrm{re}} &= (\lfloor y \rfloor + 1)\,(1/q)\,, n_{\mathrm{be}} = (\lfloor y \rfloor + (z-u)\,/v)\,(1/r), z \geq u, \end{aligned} \tag{4}$$

and we can write

$$P_{\mathrm{BL}} = \left[n_{\mathrm{re}} p + n_{\mathrm{be}}\,(1-r)\right]/L, \tag{5}$$

which is a function of p, r, and y (which depends on L). Unlike the asymptotic analysis presented above, (5) shows an explicit dependence on the frame length L. We can distinguish several cases:

- Case 1: $x \ll L$. In this case, the decoder goes through many round trips (relatively short error bursts) and the dependence on L is very slight. This gives us essentially the same result as the asymptotic analysis.
- Case 2: $x > L > 1/q$. In this case, the decoder stays in state S_{re} for an average of $1/q$ time units and then transitions to state S_{be}, remaining there for the rest of the frame. This is the error propagation case, where the dependence of P_{BL} on L is very strong. (Note that if $r = 0$, x is infinite.)
- Case 3: $L < 1/q$. In this case, the decoder never leaves state S_{re} and $P_{\mathrm{BL}} = p$.[2]

From the above analysis, we see that P_{BL} is always a decreasing function of r, as in the asymptotic case. In the absence of error propagation, we again have $r = 1 - p$, which is typically very close to 1, i.e., the probability of decoding correctly is the same whether the decoder is in state S_{re} or state S_{be}. It is precisely decoder error propagation that reduces the value of r so that the decoder typically stays in state S_{be} for a long time, and in the extreme case, i.e., low SNR, small W, and large M, r can even go to 0, which causes unlimited error propagation. In the following section, we discuss two mitigation methods designed to increase the probability r, and thus to limit the effect that decoder error propagation has on the BLER performance of SC-LDPC codes.

III. Adaptive Variable Node (VN) Doping

In order to mitigate the effect of error propagation in SWD, we introduce adaptive VN doping into the coupling chain of an SC-LDPC code. Instead of inserting doped VNs periodically into the coupling chain, as was proposed in [8], adaptive VN doping makes use of an *instantaneous noiseless binary feedback channel* to insert doped VNs into the coupling chain on an "as needed" basis, depending on the average LLR magnitudes in some number of recently decoded blocks. Before describing adaptive VN doping in more detail, we first briefly review periodic VN doping [8].

A. Periodic VN Doping

Motivated by the fact that the boundaries of a coupled chain have the effect of propagating more reliable information throughout the chain during iterative decoding, the introduction of known (or fixed) VNs into a coupled chain can emulate chain termination, thus truncating any error propagation in the iterative decoding process. Fig. 4 illustrates a periodic VN doping scheme for a (3,6)-regular SC-LDPC code, where each time unit represents a block of $2M$ coded symbols. The green VNs at times $t = \tau_1$, $\tau_2 = \tau_1 + s$, and $\tau_3 = \tau_1 + 2s$ (generally $\tau_k = \tau_1 + (k-1)s$) represent the doping points, i.e., they are fixed to known values and spaced s time units apart. The degrees of the CNs connected to these doped VNs are thus effectively reduced, introducing an irregularity into the protograph that results in stronger checks, which have the effect of truncating decoder error propagation. Unlike the case of CN doping introduced in [7], SWD of VN doped SC-LDPC codes requires no change in the decoder structure compared to undoped SC-LDPC codes.

[1] Using the model of Fig. 2, q can be calculated as $2p^2/(1+p)$, which we approximate as $2p^2$ for small p.

[2] We note that the analysis calculates the average BLER in a single decoded frame. By assuming a probability distribution for the number of time units the decoder remains in a given state, the analysis can be extended to calculate the average BLER over multiple decoded frames of a given length L [10].

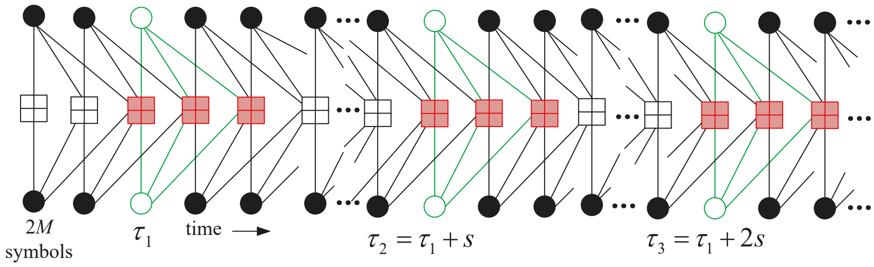

Fig. 4. VN doping for a (3,6)-regular SC-LDPC code with fixed VNs spaced periodically throughout the coupled chain.

B. Adaptive VN Doping

Unlike periodic VN doping, adaptive VN doping inserts doping points based on the average LLR magnitudes in some number of decoded blocks. In this case, (typically unequally spaced) doped VNs at times $t = \tau_1, \tau_2, \tau_3, \ldots$ are inserted into the coupling chain in response to requests from the decoder transmitted over an instantaneous noiseless binary feedback channel. To trigger a doping request in the SWD process, after completing all the iterations necessary to decode the target block at time t, if the *average decoded LLR magnitude* $\mathcal{L}_t$ satisfies

$$\mathcal{L}_t \triangleq \frac{1}{2M} \sum_{i=0}^{2M} \left|LLR_i^t\right| \leq \eta, \tag{6}$$

where η is some pre-determined *threshold*, we consider the target block at time t as *failed*. If we experience N_r consecutive failed target blocks, a doping request is submitted and the next block of VNs entering the far end of the window is assumed to be doped.

Assuming that there are d doped positions in a coupled chain, and letting n_c and n_v denote the total number of CNs and the total number of undoped VNs, respectively, the *rate* of periodic or adaptive VN doped SC-LDPC codes with frame length L and d doped VNs is given by

$$R_{\mathrm{L,doped}} = 1 - n_{\mathrm{c}}/n_{\mathrm{v}} = 1 - \left[(L+m)/(L-d)\right](1-R), \tag{7}$$

where $R = 1 - J/K$ is the *design rate* of the uncoupled LDPC-BC protograph, $d = L/s$ is fixed in the periodic case, and d is variable, depending on the frequency with which the threshold test of (6) fails N_r consecutive times, in the adaptive case. Compared to the design rate $R_{\mathrm{L}} = 1 - \left[(L+m)/L\right](1-R)$ of undoped protograph-based SC-LDPC codes [9], we see from (7) that periodic and adaptive VN doped SC-LDPC codes result in some rate loss, similar to the CN doping idea introduced in [7].

C. BLER Analysis of Periodic and Adaptive VN Doping

In order to apply the analysis of Sec. II to periodic and adaptive VN doping, we distinguish two cases: (1) a "strong code" case in which the protograph lifting factor M and the decoder window size W are both large, and (2) a "weak code" case in which M and W are both small. Based on extensive experimental observations of SWD of SC-LDPC codes, the analysis models can be assumed to have $p \ll 1$ and $r \approx 0$ for strong codes, while p is larger and r can be greater than 0 for weak codes. In other words, strong codes have very low block error rates in state S_{re} and a very small probability of reaching state S_{be}, but once they reach state S_{be} they suffer from unlimited error propagation. For this reason, very large frame lengths can suffer significant performance degradation in this case. For weak codes, on the other hand, the block error rate is much higher in state S_{re}, making them unsuitable for capacity-approaching applications, but the decoder can sometimes recover from error propagation, so large frame lengths are not necessarily catastrophic.

The VN doping methods presented earlier in this section typically result in much larger values of r. This allows the decoder to escape more quickly from state S_{be}, thus significantly improving the decoded block error rate P_{BL}, particularly at SNR operating points near capacity. For periodic VN doping, the transition probability from state S_{be} to state S_{re} increases to $r = 2/s$, where s is the spacing between doping points, since on average we must wait $s/2$ time units for a doping point. For adaptive VN doping, based on the design of Algorithm 1, the transition probability from state S_{be} to state S_{re} is given by $r = 2/(W + 2N_r - 4)$. We now calculate P_{BL}, using the asymptotic and finite L analysis models of the previous section, for both strong code and weak code examples. Based on these calculations, we then compare the predicted P_{BL} performance of undoped, periodically VN doped, and adaptively VN doped codes.

Example 1 (strong code): Choose $p = 10^{-2}$.

(a) Asymptotic ($L \to \infty$) analysis: For the undoped case ($r = 0$), the BLER is calculated using (3) as $P_{\mathrm{BL}}^{\mathrm{undoped}} = 1$. For periodic doping ($s = 200$, $r = 2/s = 10^{-2}$), we obtain $P_{\mathrm{BL}}^{\mathrm{periodic}} = 0.0196$. For adaptive doping ($W = 18$, $N_r = 2$, $r = 1/9$), we obtain $P_{\mathrm{BL}}^{\mathrm{adaptive}} = 0.0108$.

For the chosen parameters, we see that both periodic and adaptive doping significantly reduce P_{BL} compared to the undoped case, which suffers from unlimited error propagation, with adaptive doping performing about twice as well as periodic doping.

(b) Finite L analysis: Choose $L = 20,000$. For the undoped case ($r = 0$), the BLER is given by $P_{\mathrm{BL}}^{\mathrm{undoped}} = 0.505$.[3] For periodic doping ($s = 200$, $r = 10^{-2}$), using (4) and (5) we obtain $P_{\mathrm{BL}}^{\mathrm{periodic}} = 0.0149$. For adaptive doping ($W = 18$, $N_r = 2$, $r = 1/9$), we obtain $P_{\mathrm{BL}}^{\mathrm{adaptive}} = 0.0104$.

[3]When $r = 0$, the decoder stays in state S_{re} for an average of $1/q$ time units and transitions to state S_{be}, where it stays for the remaining $L - 1/q$ time units. Hence, in (5), $n_{\mathrm{re}} = 1/q$ and $n_{\mathrm{be}} = L - 1/q$.

Again, we see that both periodic and adaptive doping significantly reduce P_{BL} compared to the undoped case and that adaptive doping outperforms periodic doping.

Remark: Here we chose $r = 0$ since strong codes almost never recover from decoder error propagation, $s = 200$ since this limits the burst length to 100 on average, $W = 18 = 6v$ since this gives near optimal SWD performance at moderate-to-high SNRs, and $N_r = 2$, since anything more than single isolated errors typically indicates decoder error propagation has begun.

Example 2 (weak code): Choose $p = 10^{-1}$.

(a) Asymptotic ($L \to \infty$) analysis: For the undoped case ($r = 0.01$), the BLER is calculated using (3) as $P_{\mathrm{BL}}^{\mathrm{undoped}} = 0.524$. For periodic doping ($s = 100$, $r = 2/s = 0.02$), we obtain $P_{\mathrm{BL}}^{\mathrm{periodic}} = 0.375$. For adaptive doping ($W = 12$, $N_r = 4$, $r = 1/8$), we obtain $P_{\mathrm{BL}}^{\mathrm{adaptive}} = 0.153$.

For the chosen parameters, we see that both periodic and adaptive doping significantly reduce P_{BL} compared to the undoped case, with adaptive doping again performing better than periodic doping.

(b) Finite L analysis: Choose $L = 2,000$. For the undoped case ($r = 0.01$), the BLER is calculated using (4) and (5) as $P_{\mathrm{BL}}^{\mathrm{undoped}} = 0.545$. For periodic doping ($s = 100$, $r = 0.02$), we obtain $P_{\mathrm{BL}}^{\mathrm{periodic}} = 0.386$. For adaptive doping ($W = 12$, $N_r = 4$, $r = 1/8$), we obtain $P_{\mathrm{BL}}^{\mathrm{adaptive}} = 0.163$.

Again, we see that both periodic and adaptive doping significantly reduce P_{BL} compared to the undoped case and that adaptive doping outperforms periodic doping.

Remark: Here we chose $r = 0.01$ to reflect the fact that weak codes are less likely than strong codes to suffer from unlimited decoder error propagation, $s = 100$ since $s = 200$ would give the same result as no doping, $W = 12 = 4v$ (a weaker decoder than in Example 1), and $N_r = 4$ to reflect the fact that, for weak codes, we must wait longer before declaring that error propagation has begun and sending a doping request.

Finally, we note that, either with or without doping, strong codes perform much better than weak codes due to the smaller value of p, since strong codes are much more resilient to channel errors. Also, for smaller values of L, the gains achieved by doping are expected to be less dramatic, since error propagation is not as damaging for small frame lengths.

IV. Numerical Results

In order to verify the effectiveness of the proposed adaptive VN doping scheme, the BLER performance of the undoped, periodically doped, and adaptively doped (3,6)-regular SC-LDPC codes of Fig. 1 with SWD is shown in Fig. 5, assuming an AWGN channel with BPSK signaling, $M = 2000$, $W = 12$, $N_r = 2$, and $L = 1000$, where we use 2 points for periodic doping and a maximum of 2 doping points for adaptive doping.[4] The results confirm our analysis in Sec. III that, when low latency operation is desired at the lower SNRs typically used in practice, code doping significantly improves the BLER performance, with adaptive doping outperforming periodic doping.

[4]We limit the number of doping points for adaptive doping so that its rate loss can never exceed that of periodic doping.

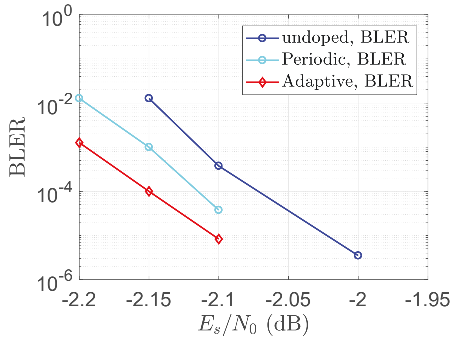

Fig. 5. Performance comparison of undoped, periodically doped, and adaptively doped (3,6)-regular SC-LDPC codes.

V. Conclusions

In this paper, we proposed a general decoder model for SWD of SC-LDPC codes subject to infrequent but severe decoder error propagation. We then introduced an adaptive VN doping strategy to combat the error propagation. Finally, using both an analysis based on the decoder model and simulation results, we showed that adaptive VN doping outperforms both undoped codes and a periodic VN doping strategy.

Acknowledgment

This material is based upon work supported by the National Science Foundation under Grant Nos. OIA-1757207 and HRD-1914635.

References

[1] A. J. Felström, and K. Sh. Zigangirov, "Time-varying periodic convolutional codes with low-density parity-check matrix," *IEEE Trans. Inf. Theory*, vol. 45, no. 6, pp. 2181-2191, Sep. 1999.

[2] M. Lentmaier, A. Sridharan, D. J. Costello, Jr., and K. Sh. Zigangirov, "Iterative decoding threshold analysis for LDPC convolutional codes," *IEEE Trans. Inf. Theory*, vol. 56, no. 10, pp. 5274-5289, Oct. 2010.

[3] S. Kudekar, T. J. Richardson, and R. L. Urbanke, "Threshold saturation via spatial coupling: Why convolutional LDPC ensembles perform so well over the BEC," *IEEE Trans. Inf. Theory*, vol. 57, no. 2, pp. 803-834, Feb. 2011.

[4] A. R. Iyengar, M. Papaleo, P. H. Siegel, J. K. Wolf, A. Vanelli-Coralli, and G. E. Corazza, "Windowed decoding of protograph-based LDPC convolutional codes over erasure channels," *IEEE Trans. Inf. Theory*, vol. 58, no. 4, pp. 2303-2320, Apr. 2012.

[5] K. Klaiber, S. Cammerer, L. Schmalen, and S. ten Brink, "Avoiding burst-like error patterns in windowed decoding of spatially coupled LDPC codes," in *Proc. IEEE 10th Int. Symp. on Turbo Codes & Iterative Inf. Processing*, Hong Kong, China, Dec. 3-7, 2018, pp. 1-5.

[6] M. Zhu, D. G. M. Mitchell, M. Lentmaier, D. J. Costello and B. Bai, "Error Propagation Mitigation in Sliding Window Decoding of Braided Convolutional Codes," in IEEE Transactions on Communications, doi: 10.1109/TCOMM.2020.3015945, 2020.

[7] M. Zhu, D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, Jr., "A novel design of spatially coupled LDPC codes for sliding window decoding," in *Proc. IEEE Int. Symp. Information Theory*, Los Angeles, CA, USA, June 21-26, 2020, pp. 473-478.

[8] M. Zhu, D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, Jr., "Decoder error propagation mitigation for spatially coupled LDPC codes," to appear in *International Symposium on Information Theory and Its Applications*, Kapolei, Hawai'i, USA, October 24-27, 2020.

[9] D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, Jr., "Spatially coupled LDPC codes constructed from protographs," *IEEE Trans. Inf. Theory*, vol. 61, no. 9, pp. 4866-4889, July 2015.

[10] M. Zhu, D. G. M. Mitchell, M. Lentmaier, and D. J. Costello, Jr., "Code doping techniques for combating error propagation in sliding window decoding of spatially coupled LDPC codes," to be submitted to *IEEE Trans. Inf. Theory*.