Ultra-Reliable Wireless Communications via Incremental Redundancy and Space-Time Coding

Kaiming Shen[†], Saeed R. Khosravirad[‡], and Wei Yu[†]
[†]Electrical and Computer Engineering Dept., University of Toronto, ON M5S 3G4, Canada
[‡]Nokia Bell Labs, Murray Hill, NJ
E-mail: kshen@ece.utoronto.ca, saeed.khosravirad@nokia-bell-labs.com, weiyu@ece.utoronto.ca

***Abstract*—Multi-hop relaying has emerged as a key technique for ultra-reliable communications due to its ability to harness spatial location diversity while avoiding interference. In this framework, the user messages are aggregated into a single packet and broadcast to all the receivers in multiple phases, thus allowing the stronger receivers to relay the packet to the weaker receivers. Prior implementations of the above framework, however, typically do not leverage the past received signals to improve message decoding. This paper proposes using incremental redundancy coding with coordinated decoding across the multiple phases, in conjunction with transmission using space-time coding, to enable higher achievable rate overall. The resulting scheme provides significant improvement over the existing methods. Moreover, we propose an extension that incorporates multi-cell interference management.**

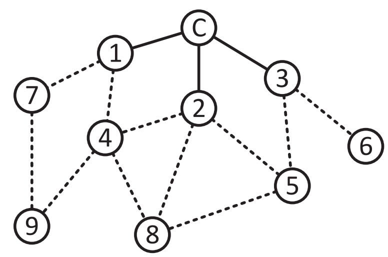

Fig. 1. The controller (C) broadcasts a common message to nine actuators using multi-hop relaying. Nodes 1 to 3 are able to decode the message in the first phase, nodes 4 to 7 in the second phase, and the rest of the nodes in the third phase.

I. Introduction

Ultra-reliable communications are motivated by a number of evolving applications such as industrial automation, vehicle-to-vehicle wireless, artificial intelligence in healthcare, and virtual reality, all of which demand a much lower target packet error probability than the current wireless systems. As shown in [1], [2], deep fading and interference pose a major challenge to the practical realization of ultra-reliability. Inspired by a multi-hop framework in [3], this work proposes the idea of combining *incremental redundancy* and *space-time coding* to facilitate ultra-reliable communications in the presence of the above challenge.

Consider a network topology as shown in Fig. 1, modeling an automated production line with a controller (denote as C) and a number of actuators. The controller wishes to send a set of independent control messages to the actuators within a fixed number of transmission phases. Following the previous works [3], [4], we aggregate all these individual messages into a common message that is intended for every actuator. This message aggregation, albeit scaling down the code rate, benefits the transmission reliability in two aspects. First, it eliminates interference since every receiver now desires the same message. Second, those actuators that already decode the message successfully can help relay the packet toward the remaining actuators.

Although the same framework has been considered in [3], [4], this work differs from the previous studies in the way the decoding is coordinated across the phases. What has been proposed in [3], [4] is basically an uncoordinated scheme wherein the receiver tries to decode the message *solely* based on the current transmission phase, with the past packets discarded. In contrast, the proposed scheme in this paper utilizes all the received packets *jointly* using incremental redundancy coding. As a result, the maximum reliable data rate allowed by the proposed method can be significantly higher than that allowed by the previous methods [3], [4]. Furthermore, we compare the proposed method with another way of coordinating phases by means of message splitting [5]. Unlike the incremental redundancy method, the message-splitting approach requires channel state information at transmitter (CSIT) but does not require signal buffering. The gap between the achievable rates of the two methods is analyzed for the two-phase case.

A sequence of recent works [4], [6], [7] propose various improvements beyond the prototype multi-hop framework originally proposed in [3], but none of them consider coordinated decoding across the phases. The main idea of multi-hop relaying is to exploit the spatial diversity arising from the actuator locations. Other types of diversity have been considered as well in the existing literature [8], [9], but as pointed out in [3], the spatial location diversity is far more reliable in practice than the time diversity (which depends on the very small coherence time) and the frequency diversity (which depends on sufficient multi-path). This work does not consider the effect of blocklength, but we mention here several work [10], [11] on coding over short packets in the context of ultra-reliable low-latency communications (URLLC).

II. System Model and Proposed Protocol

Consider an automated production line with one controller and a set of remote actuators $\mathcal{K} = \{1, 2, \ldots, K\}$. We refer to the area occupied by this production line as *cell*. The role of the controller is to wirelessly stream a control message to each of the associated actuators in the cell. Independent control message of size b bits needs to be received at each actuator within period T. We assume that T is less than the coherence time, so the channels are all fixed in our discussion. Let $g_{kc} \in \mathbb{C}$ be the channel from the controller to the kth actuator, and $g_{k\ell} \in \mathbb{C}$ the channel from the ℓth actuator to the kth actuator. Due to deep fading and interference, not every actuator has a sufficiently strong wireless link from the controller. The work [3] proposes a multi-hop relaying framework to enhance the transmission reliability as described below.

The time interval $[0, T)$ is equally partitioned into D phases: $[0, T/D), [T/D, 2T/D), \ldots, [(D-1)T/D, T)$. Here and throughout, we use the superscript d to denote which phase a particular variable is associated with. In the first phase, the controller is the only transmitting node, so each actuator $k \in \mathcal{K}$ receives

$$Y_k^{(1)}(t) = g_{kc} X_c^{(1)}(t) + Z_k^{(1)}(t), \tag{1}$$

for $t \in [0, T/D)$, where $X_c^{(1)}$ is the signal transmitted by the controller under the power constraint $\mathbb{E}\big[\big|X_c^{(1)}\big|^2\big] \le p_c$, and the complex Gaussian random variable $Z_k^{(1)} \sim \mathcal{CN}(0, \sigma^2)$ is the background noise.

Let $\mathcal{A}_d \subseteq \mathcal{K}$ be the set of actuators that have already successfully decoded the packet prior to the dth phase, and let $\bar{\mathcal{A}}_d$ be its complement. In the dth phase, those actuators in $\mathcal{A}_d$ would assist the controller by cooperatively transmitting the message to the remaining actuators in the cell, so each actuator $k \in \bar{\mathcal{A}}_d$ would receive

$$Y_k^{(d)}(t) = g_{kc} X_c^{(d)}(t) + \sum_{\ell \in \mathcal{A}_d} g_{k\ell} X_\ell^{(d)}(t) + Z_k^{(d)}(t), \tag{2}$$

for $t \in \big[(d-1)\frac{T}{D}, d\frac{T}{D}\big)$, where $X_c^{(d)}$ is the signal transmitted by the controller, $X_\ell^{(d)}$ is the signal transmitted by the ℓth actuator, and $Z_k^{(d)} \sim \mathcal{CN}(0, \sigma^2)$ is the background noise. Power constraint is imposed on each transmitter such that $\mathbb{E}\big[\big|X_c^{(d)}\big|^2\big] \le p_c$ and $\mathbb{E}\big[\big|X_k^{(d)}\big|^2\big] \le p_k$, $k \in \mathcal{K}$, in each phase.

For ease of notation, the time variable t is omitted in the rest of the paper unless otherwise stated. Introduce the notation

$$\gamma_c = \frac{p_c}{\sigma^2} \quad \text{and} \quad \gamma_k = \frac{p_k}{\sigma^2}. \tag{3}$$

We also use $\mathsf{C}(a)$ as a shorthand for $\log_2(1+a)$.

The multi-hop relaying framework lets the controller aggregate all its K independent messages into a single bK-bit message m, and further requires all the actuators in the cell to decode m within the D phases. Hence, the target rate for every actuator is

$$\hat{R} = \frac{bK}{T}. \tag{4}$$

III. Existing Strategy: Occupy CoW

The idea of the Occupy CoW method [3], [4] is fairly simple: the controller broadcasts the message m in the first phase, then repeats m along with the successful actuators in the later phases. Hence, the signal-to-interference-plus-noise ratio (SINR) of actuator k in the first phase is

$$s_k^{(1)} = |g_{kc}|^2 \gamma_c. \tag{5}$$

The decoding of actuator k is successful if $\frac{1}{D}\mathsf{C}(s_k^{(1)}) \ge \hat{R}$, and fails otherwise. Furthermore, the SINR of the actuator $k \in \bar{\mathcal{A}}_d$ in phase d is

$$s_k^{(d)} = |g_{kc}|^2 \gamma_c + \sum_{\ell \in \mathcal{A}_d} |g_{k\ell}|^2 \gamma_\ell. \tag{6}$$

Similarly, its decoding in phase d is successful if and only if $\frac{1}{D}\mathsf{C}(s^{(d)}) \ge \hat{R}$. If it still fails in the last phase, i.e.,

$$\frac{1}{D} \cdot \mathsf{C}(s_k^{(D)}) < \hat{R}, \tag{7}$$

then the actuator would not be able to decode m. We remark that the Occupy CoW method does not assume CSIT, but the receiver still needs to estimate its channel from the controller and the relay actuator(s) through pilot signaling.

The Occupy CoW method can be quite inefficient in the sense that actuator k would encounter a complete decoding failure in phase d even if its $\frac{1}{D}\mathsf{C}(s_k^{(d)})$ is only slightly below $\hat{R}$. Our new design aims to address this issue by using incremental redundancy coding and coordinated decoding based on the received signals from the multiple phases.

IV. Proposed Method Based on Incremental Redundancy and Space-Time Coding

This section consists of three parts. It begins with the coding strategy for multi-hop relaying and presents a vector channel interpretation that directly yields the achievable rate. The second part of this section discusses a practical achievability based on space-time coding and incremental redundancy. The final part gives an analytical comparison between the proposed incremental redundancy method and the message splitting method from [5].

A. Achievable Rate via Vector Channel Representation

In the multi-hop relaying framework discussed in Section II, those actuators that have already recovered the message m serve as relays for the rest of the actuators in subsequent phases. It is therefore natural to adopt a *decode-and-forward* relaying scheme. However, it is crucial to note that if the decoder is to utilize the received signals from multiple phases jointly for decoding the message m, then the relays should not simply re-use the same codebook and re-broadcast the same coded message—doing so would merely provide an SNR gain, while a much better strategy is to re-encode the message m using different and independently generated codebooks.

In this section, we use a random coding argument to derive the overall achievable rate for the multi-hop relaying scheme. We propose to generate different codebooks independently for every node and for every phase, i.e., for each (ℓ, d) or (c, d) pair. Within each codebook, the codeword given m is randomly generated according to the complex Gaussian distribution $\mathcal{CN}(0, p_\ell)$ or $\mathcal{CN}(0, p_c)$ in an i.i.d. fashion.

To derive the achievable rate using such a coding strategy, we write the transmit signal over the multiple phases in (1) and (2) as follows. Let $X^{(d)}$ be the set of signals transmitted in phase d, *i.e.,*

$$X^{(d)} = (X_c^{(d)}, X_1^{(d)}, \ldots, X_K^{(d)}), \tag{8}$$

and further let

$$\mathbf{X}^{(d)} = (X^{(1)}, X^{(2)}, \ldots, X^{(d)})^T, \tag{9}$$

where $X_\ell^{(d)} = 0$ if actuator $\ell \in \bar{\mathcal{A}}_d$. Likewise, for the actuator $k \in \bar{\mathcal{A}}_d$, the received signal vector and the noise vector over the first d phases are expressed as

$$\mathbf{Y}_k^{(d)} = (Y_k^{(1)}, Y_k^{(2)}, \ldots, Y_k^{(d)})^T \tag{10}$$

and

$$\mathbf{Z}_k^{(d)} = (Z_k^{(1)}, Z_k^{(2)}, \ldots, Z_k^{(d)})^T, \tag{11}$$

respectively. Denoting the channels related to actuator k as

$$\mathbf{g}_k = (g_{kc}, g_{k1}, \ldots, g_{kK}), \tag{12}$$

we construct the following $d \times Kd$ channel matrix:

$$\mathbf{G}_k^{(d)} = \begin{bmatrix} \mathbf{g}_k & & & \\ & \mathbf{g}_k & & \\ & & \ddots & \\ & & & \mathbf{g}_k \end{bmatrix}_{d \times Kd}. \tag{13}$$

The scalar channels of (1) and (2) can then be combined into a virtual vector channel:

$$\mathbf{Y}_k^{(d)} = \mathbf{G}_k^{(d)} \mathbf{X}^{(d)} + \mathbf{Z}_k^{(d)}. \tag{14}$$

Now assuming that we use independently generated codebooks across the d phases and across all the relaying node, then by joint decoding across the d phases, the following rate is achievable:

$$\begin{aligned} R_k^{(d)} &= \frac{1}{D} \log_2 \left| \boldsymbol{I}_d + \frac{1}{\sigma^2} \mathbf{G}_k^{(d)} \mathbb{E}\Big[\mathbf{X}^{(d)} (\mathbf{X}^{(d)})^H\Big] (\mathbf{G}_k^{(d)})^H \right| \\ &= \frac{1}{D} \cdot \sum_{q=1}^{d} \mathsf{C}\left(|g_{kc}|^2 \gamma_c + \sum_{\ell \in \mathcal{A}_q} |g_{k\ell}|^2 \gamma_\ell \right). \end{aligned} \tag{15}$$

This is in contrast to the Occupy CoW method [3], [4], which suggests independent decoding across the phases, yielding the data rate of

$$R_k^{(d)} = \frac{1}{D} \cdot \mathsf{C}\left(|g_{kc}|^2 \gamma_c + \sum_{\ell \in \mathcal{A}_d} |g_{k\ell}|^2 \gamma_\ell \right). \tag{16}$$

Observe that our achievable rate is significantly higher, due to joint decoding and the use of independent codebooks.

B. Practical Coding Strategy

The rate expression in (15) is obtained from a random coding argument. This section aims to provide an incremental redundancy and space-time coding strategy for achieving this rate. The proposed approach provides more insights into the practical code construction for achieving (15).

We start with the space-time coding. Given the signal received in phase d alone, the data rate

$$r_k^{(d)} = \frac{1}{D} \cdot \mathsf{C}\left(|g_{kc}|^2 \gamma_c + \sum_{\ell \in \mathcal{A}_d} |g_{k\ell}|^2 \gamma_\ell \right) \tag{17}$$

is achievable for actuator $k \in \bar{\mathcal{A}}_d$ by using the distributed space-time coding among the controller and the relay actuators in $\mathcal{A}_d$. Importantly, the space-time coding can be readily extended to the multi-receiver case. From a mutual information perspective, there exist a set of codebooks $\mathcal{C}_c^{(d)}$ and $\mathcal{C}_\ell^{(d)}$ such that

$$r_k^{(d)} = \frac{1}{D} \cdot I(X^{(d)}; Y_k^{(d)}) \tag{18}$$

is achievable for actuator $k \in \bar{\mathcal{A}}_d$. The use of space-time coding for multi-hop relaying already appears in the previous works [3], [4].

We proceed to describe incremental redundancy. Below is a classic result of incremental redundancy for an isolated link.

Lemma 1 (Lemma 1 of [12]): Consider a point-to-point channel in which the transmitter aims to send some message m to the receiver in D phases, each of the blocklength L. There exists an incremental redundancy scheme for transmitting the message $m \in [1 : 2^{LR}]$ reliably as long as

$$\frac{1}{D} \cdot \sum_{d=1}^{D} I(X^{(d)}; Y^{(d)}) \geq R, \tag{19}$$

where $X^{(d)}$ and $Y^{(d)}$ refer to the channel input and the channel output in phase d, respectively.

The above result can be readily extended to a broadcast scenario with multiple receivers.

Proposition 1: Assume that the transmitter aims to send some common message m to a set of K receivers after D phases. There exists an incremental redundancy scheme for transmitting the common message $m \in [1 : 2^{LR}]$ reliably provided that

$$\min_{k \in [1:K]} \left\{ \frac{1}{D} \cdot \sum_{d=1}^{D} I(X^{(d)}; Y_k^{(d)}) \right\} \geq R, \tag{20}$$

where $X^{(d)}$ and $Y_k^{(d)}$ refer to the channel input and the channel output related to receiver k in phase d, respectively.

The idea of incremental redundancy originates from the *Hybrid-ARQ* [12], *i.e.*, the transmitter would send additional blocks carrying new incremental redundancy bits upon the request from the receiver.

We now return to our channel. Using the space-time coding among the transmitters for each phase, we further coordinate the different phases via incremental redundancy coding as in Proposition 1, thereby achieving

$$R_k^{(d)} = \sum_{q=1}^{d} r_k^{(q)}, \tag{21}$$

which equals to (15) exactly. The coding strategy is outlined below:

TABLE I
COMPARISON OF THE DIFFERENT IMPLEMENTATIONS OF THE MULTI-HOP RELAYING FRAMEWORK.

	Occupy CoW [3], [4]	Message Splitting [5]	Incremental Redundancy
Signal Buffering	Not Needed	Not Needed	Required
CSIT	Not Needed	Required	Not Needed
Decoding Scheme	Independent in Each Phase	SIC	Joint Decoding Across Phases

1) Initialization: set $d = 1$;
2) In phase d, the controller and the relays in $\mathcal{A}_d$ encode the message m using the codebook $\mathcal{C}^{(d)}$;
3) After encoding, the controller and the relays coordinate their signals via distributed space-time coding;
4) On the receiver side, each actuator $k \in \bar{\mathcal{A}}_d$ tries to recover m based on the current signal $Y_k^{(d)}$ and the past signals $Y_k^{(1)}, \ldots, Y_k^{(d-1)}$ jointly;
5) Those actuators that successfully recover m are added to the relay set $\mathcal{A}_{d+1}$; the rest actuators read the current packet into their buffers;
6) Update $d = d + 1$; go back to step 2 unless all the actuators have recovered m successfully or the maximum number of iterations is reached.

A few comments are in order about the *codebooks* and the *joint decoding* aspect of the above algorithm. We propose to use the incremental redundancy low-density parity-check (IR-LDPC) codes. In this strategy, the first codebook $\mathcal{C}^{(1)}$ is used to encode the message m, while the later codebooks $\mathcal{C}^{(d)}$, $d = 2, \ldots, D$, provide parity-check bits for the codeword from $\mathcal{C}^{(1)}$. Hence, in step 4, actuator k collects all the parity check bits obtained so far, then uses them jointly to decode m. Fig. 2 illustrates this multi-hop relaying protocol.

C. Incremental Redundancy vs. Message Splitting

In a previous work [5], we propose a different way of coordinating the different transmission phases via message splitting so that the receiver can partially decode m even if its current achievable rate is below the target rate $\hat{R}$. This section illustrates the difference between the incremental redundancy and the message-splitting schemes, using $D = 2$ case as an example (so there are two phases in total).

The message splitting method [5] partitions the message m into m' and m''. Assume that a fraction $0 \leq \lambda \leq 1$ of the total transmit power is allocated to m' and the remaining power is allocated to m''. The main idea is to ensure that every actuator can at least recover m' in phase 1; the strong actuators can recover m'' in phase 1 as well, while the weak ones recover m'' in phase 2. This message splitting method [5] is specified as follows:

1) In phase 1, the controller broadcasts (m', m'');
2) Each actuator tries to decode m' and then m'' by successive interference cancellation (SIC);
3) In phase 2, the controller works with the actuators that recover the pair (m', m'') in transmitting m'' via distributed space-time coding;
4) The remaining actuators try to decode m''.

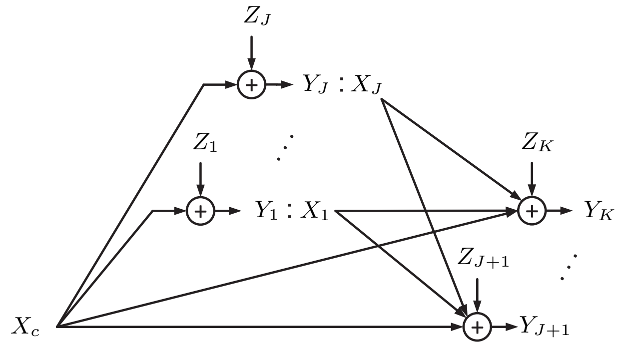

Fig. 2. Consider using the incremental redundancy method in a two-phase system. Assume that the actuators $1, \ldots, J$ recover the message m successfully in phase 1. They help relay m to the remaining actuators $J+1, \ldots, K$ in phase 2.

Hence, there are two types of decoding error: (i) some actuator fails to recover m' in phase 1; (ii) some actuator fails to recover m'' in both two phases.

The performance of the message splitting method is sensitive to the power and rate splitting ratio λ, which must be chosen judiciously at the transmitter side according to the channel condition, so CSIT is required. However, unlike the incremental redundancy method, the message splitting method does not require the actuator to buffer the past signals. Table I summarizes the difference between incremental redundancy and message splitting schemes.

The main result of this section is the following comparison in the achievable rates of the two methods.

Proposition 2: Let R be the achievable rate of the incremental redundancy method in (15), and let $\tilde{R}$ be the achievable rate of the message splitting method. When $D = 2$, these two data rates always satisfy

$$R - \delta \leq \tilde{R} \leq R, \tag{22}$$

where

$$\delta = \frac{1}{2}\log_2\left(\frac{\max_{\ell\in\bar{\mathcal{A}}_1}|g_{\ell c}|^2}{\min_{\ell\in\bar{\mathcal{A}}_1}|g_{\ell c}|^2}\right) + \frac{1}{2}. \tag{23}$$

Corollary 1: If all the actuators in $\bar{\mathcal{A}}_d$ (*i.e.,* the weak actuators) have equal channel strength from the controller, then the gap $\delta = 0.5$.

The rest of this section aims to prove Proposition 2. To ease notation, we assume without loss of generality that $|g_{1c}| \geq |g_{2c}| \geq \cdots \geq |g_{Kc}|$. First, let us consider the incremental redundancy method and assume that actuators 1 to J are able to recover m after phase 1. This setup is illustrated in Fig. 2.

The resulting achievable rate of the incremental redundancy method is

$$R=\frac{1}{2}\min\left\{\mathsf{C}(|g_{Jc}|^2\gamma_c),\min_{\ell\in[J+1:K]}\mathsf{C}(|g_{\ell c}|^2\gamma_c)+\mathsf{C}(\eta_\ell)\right\}, \tag{24}$$

where

$$\eta_\ell=|g_{\ell c}|^2\gamma_c+\sum_{i=1}^{J}|g_{\ell i}|^2\gamma_i. \tag{25}$$

Now, consider rate-splitting. Let R' and R'' be the data rates of m' and m'', respectively, so the total achievable rate is $\tilde{R}=R'+R''$. Since $\tilde{R}\le R$ is straightforward, we focus on proving the other inequality of (22). For the message splitting method, we propose to set

$$\lambda=\max\left\{0,1-\frac{1}{|g_{Kc}|^2\gamma_c}\right\}. \tag{26}$$

Recall that the submessage m' would be recovered by every actuator in phase 1, so its maximum rate is bounded as

$$\begin{aligned}R'&\ge\min_{\ell\in[1:K]}\frac{1}{2}\mathsf{C}\left(\frac{|g_{\ell c}|^2\gamma_c}{1+|g_{\ell c}|^2/|g_{Kc}|^2}\right)\\&=\frac{1}{2}\mathsf{C}\big(|g_{Kc}|^2\gamma_c\big)-\frac{1}{2}.\end{aligned} \tag{27}$$

Suppose that actuators 1 to $\tilde{J}$ are able to recover (m',m'') by SIC in phase 1. Since $R'+R''\le R$, we have $\tilde{J}\ge J$. Consequently, the rate of m'' can be bounded as

$$\begin{aligned}R''&=\min\left\{R-R',\min_{\ell\in[\tilde{J}+1:K]}\frac{1}{2}\mathsf{C}\left(|g_{\ell c}|^2\gamma_c+\sum_{i=1}^{\tilde{J}}|g_{\ell i}|^2\gamma_i\right)\right\}\\&\ge\min\left\{R-R',\min_{\ell\in[J+1:K]}\frac{1}{2}\mathsf{C}(\eta_\ell)\right\}.\end{aligned} \tag{28}$$

Integrating the above result in $\tilde{R}=R'+R''$, we obtain

$$\begin{aligned}R-\tilde{R}\le&\min_{\ell\in[J+1:K]}\left\{\frac{1}{2}\mathsf{C}(|g_{\ell c}|^2\gamma_{\ell c})+\frac{1}{2}\mathsf{C}(\eta_\ell)\right\}\\&-\frac{1}{2}\mathsf{C}(|g_{Kc}|^2\gamma_c)-\min_{\ell\in[j+1:K]}\frac{1}{2}\mathsf{C}(\eta_\ell)+\frac{1}{2}\\\le&\min_{\ell\in[J+1:K]}\left\{\frac{1}{2}\mathsf{C}(|g_{\ell c}|^2\gamma_{\ell c})-\frac{1}{2}\mathsf{C}(|g_{Kc}|^2\gamma_c)\right\}+\frac{1}{2}\\\le&\frac{1}{2}\log_2\left(\frac{|g_{J+1,c}|^2}{|g_{K,c}|^2}\right)+\frac{1}{2}.\end{aligned} \tag{29}$$

The proof of Proposition 2 is thus completed.

V. Extension

We further consider the multi-cell case. The earlier work [3] considers using orthogonal spectrum bands across the cells in order to avoid inter-cell interference. However, this strategy can be problematic in that its total bandwidth requirement grows linearly with the number of cells. The more recent work [4] proposes to let the actuators perform SIC on the signals from the nearby cells. The authors of [4] put forward two schemes: (i) Using SIC for every phase; (ii) using SIC in phase 1 alone and then the orthogonal division in the remaining phases. Through extensive numerical results, it is shown that the second scheme is superior when the signal-to-noise ratio (SNR) is higher than 0 dB. Thus, the second scheme of [4] is more suited in our case. Besides, the authors of [4] argue that using orthogonal division in the second phase can improve the diversity order. For the multi-cell system, we thus propose to replace the Occupy CoW method with the incremental redundancy method in the second scheme of [4].

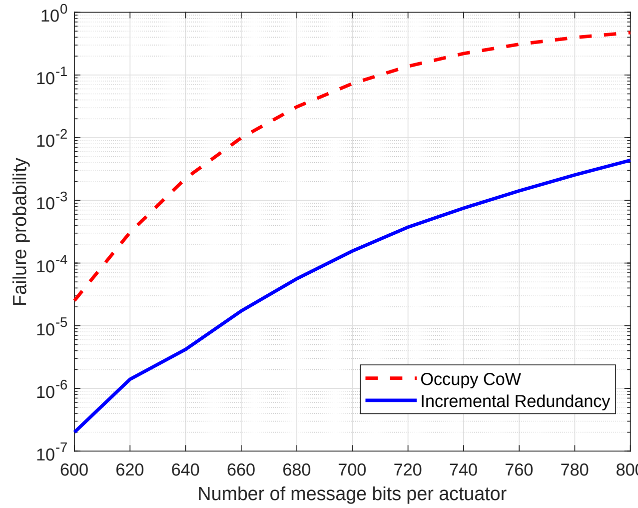

Fig. 3. Actuator failure probability in the single-cell case.

VI. Numerical Examples

The performance of the proposed approach is validated in a typical environment setup [4] as follows. Given a pair of transmitter and receiver that are L meters apart from each other, the channel pathloss (in dB) between them is computed differently for two cases: it equals to $18.7\log_{10}(L)+46.8+20\log_{10}(0.6)$ if the channel is line-of-sight (LOS), and equals to $36.8\log_{10}(L)+46.8+20\log_{10}(0.6)$ if the channel is non-line-of-sight (NLOS). We assume that the channel must be LOS when $L\le 2.5$ m, and would be LOS with a probability of $(1-0.9(1-(1.24-90.61\log_{10}(L))^3)^{1/3}$ otherwise. This implies that deep fading is more likely when the transmitter and the receiver are far apart. Moreover, the shadowing effect is modeled as a Gaussian variable $\mathcal{N}(0,4)$ in the decibel scale. We set the total spectrum bandwidth to 5 MHz, and set the total period T to 1 ms in order to account for low latency. Assume also that each cell is a 10 m $\times$ 10 m square area with the controller located at the centre and the associated actuators uniformly distributed. We set $D=2$, $K=30$, and $p_c=p_k=5$ dBm.

We first consider the single-cell case. The proposed incremental redundancy scheme as stated in Section IV-B is compared with the existing method named “Occupy CoW” in [3]. (Note that the method of [4] reduces to Occupy CoW in the single-cell case.) Fig. 3 compares the actuator failure probabilities of the two methods for the different control

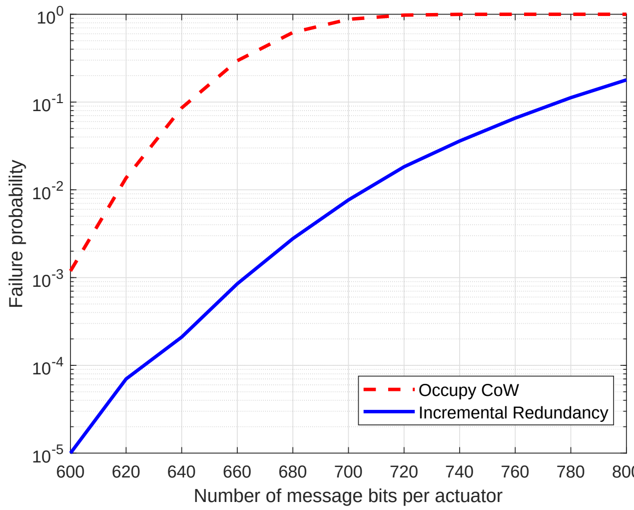

Fig. 4. System failure probability in the single-cell case.

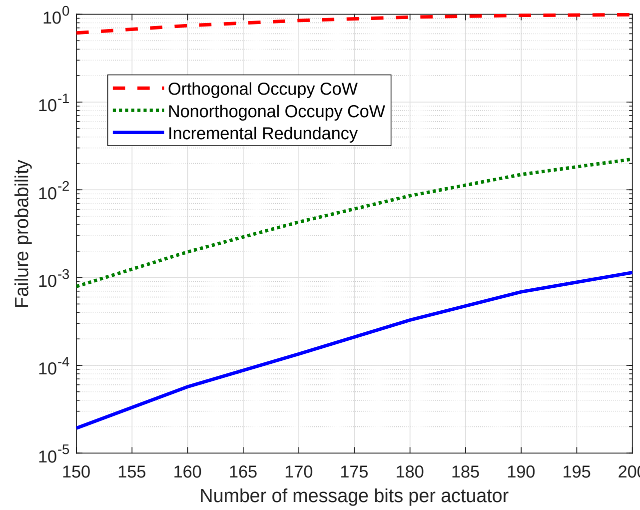

Fig. 5. Actuator failure probability in the multi-cell case.

message sizes b. According to the figure, the proposed method outperforms Occupy CoW significantly, *e.g.*, the gain is over 25 dB at $b = 680$ bits. Observe also that the actuator failure probability of the proposed method increases with b more slowly. Fig. 4 compares the system failure probability—the probability of at least one actuator failing to decode the message. The improvement of the proposed method over Occupy CoW is still substantial. In particular, when $b = 700$ bits, Occupy CoW almost surely encounters a system failure, whereas the proposed method suppresses the system failure probability below 1%.

The remainder of this section considers multiple cells. Assume that a total of 9 cells (each with 30 actuators) are located in a 3×3 grid. The distance between any two adjacent controllers is 30 m.

We compare the proposed multi-cell method of Section V with Orthogonal Occupy CoW [3] (that is based on the orthogonal division) and Nonorthogonal Occupy CoW [4] (that is based on SIC for the first phase and orthogonal division for the second phase). As shown in Fig. 5, Orthogonal Occupy CoW has disastrous performance; its actuator failure probability is always much higher than 1%. The proposed method outperforms Nonorthogonal Occupy CoW significantly; the difference between the two methods is more than 10 dB.

VII. Conclusion

This paper proposes a novel strategy for ultra-reliable communications in an industrial automation environment. Incremental redundancy and space-time coding are the two building blocks of the proposed method. As opposed to the existing multi-hop relaying schemes that discard the past received signals, the proposed strategy makes better use of diversity by coordinated decoding across the multiple phases, thus significantly enhancing the reliability. We also analytically compare this new method with the message splitting method. Further, we extend the proposed incremental redundancy method to account for the multi-cell scenario with inter-cell interference. Simulations demonstrate significant advantages of the proposed method against the benchmarks in improving the reliability of wireless communications.

References

[1] P. Popovski, "Ultra-reliable communication in 5G wireless systems," in *Int. Conf. 5G Ubiquitous Connectivity*, Nov. 2014, pp. 26–28.

[2] O. N. C. Yilmaz, Y.-P. Wang, N. A. Johansson, N. Brahmi, and *et al.*, "Analysis of ultra-reliable and low-latency 5G communication for a factory automation use case," in *IEEE Int. Conf. Commun. Workshop (ICCW)*, June 2015, pp. 1190–1195.

[3] V. N. Swamy, S. Suri, P. Rigge, M. Weiner, G. Ranade, A. Sahai, and B. Nikolić, "Real-time cooperative communication for automation over wireless," *IEEE Trans. Wireless Commun.*, vol. 16, no. 11, pp. 7168–7183, Nov. 2017.

[4] S. A. Ayoughi, W. Yu, S. R. Khosravirad, and H. Viswanathan, "Interference mitigation for ultrareliable low-latency wireless communication," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 4, pp. 869–880, Apr. 2019.

[5] K. Shen, W. Yu, and S. R. Khosravirad, "Ultrareliable wireless communication with message splitting," in *IEEE Int. Workshop Sig. Process. Advances Wireless Commun. (SPAWC)*, July 2019.

[6] L. Liu and W. Yu, "A D2D-based protocol for ultra-reliable wireless communications for industrial automation," *IEEE Trans. Wireless Commun.*, vol. 17, no. 8, pp. 5045–5058, Aug. 2018.

[7] V. N. Swamy, P. Rigge, G. Ranade, B. Nikolić, and A. Sahai, "Wireless channel dynamics and robustness for ultra-reliable low-latency communications," *IEEE J. Sel. Areas Commun.*, vol. 37, no. 4, pp. 705–720, Apr. 2019.

[8] N. A. Johansson, Y.-P. Wang, E. Eriksson, and M. Hessler, "Radio access for ultra-relaible and low-latency 5G communications," in *IEEE Int. Conf. Commun. Workshop (ICCW)*, June 2015, pp. 1184–1189.

[9] P. Popovski, J. J. Nielsen, Č. Stefanovió, E. de Carvalho, and *et al.*, "Wireless access for ultra-reliable low-latency communications: Principles and building blocks," *IEEE Netw.*, vol. 32, no. 2, pp. 16–23, Apr. 2018.

[10] M. Sybis, K. Wesołowski, K. Jayasinghe, and V. Venkatasubramanian, "Channel coding for ultra-relaible low-latency communication in 5G systems," in *IEEE Veh. Tech. Conf. (VTC-Fall)*, Sept. 2016.

[11] G. Durisi, T. Koch, and P. Popovski, "Toward massive, ultrareliable, and low-latency wireless communication with short packets," *Proc. IEEE*, vol. 104, no. 9, pp. 1711–1726, Sept. 2016.

[12] G. Caire and D. Tuninetti, "The throughput of hybrid-ARQ protocols for the Gaussian collision channel," *IEEE Trans. Inf. Theory*, vol. 47, no. 5, pp. 1971–1988, July 2001.