Wireless Video Multicast with Cooperative and Incremental Transmission of Parity Packets

Zhili Guo, Student Member, IEEE, Yao Wang, Fellow, IEEE, Elza Erkip, Fellow, IEEE,
Shivendra Panwar, Fellow, IEEE,

Abstract—This paper introduces a novel and efficient approach for user cooperation in wireless video multicast using Randomized Distributed Space Time Codes (R-DSTC), in which the sender first transmits the source packets, and the sender and receivers that have received all source packets then generate and send the parity packets simultaneously using R-DSTC. As more parity packets are delivered, more receivers can recover all source packets and join the parity packet transmission. Four variations of the proposed systems are considered. The first one requires complete channel information between the sender and all receivers and between all receivers to derive the optimal transmission rates for sending source and parity packets, and employs receiver feedback to determine when to terminate parity transmission. The other three suboptimal systems do not require full channel information and/or receiver feedback, and hence are more feasible in practice. All four versions can support significantly higher video rates and correspondingly higher quality of decoded video, than prior approach in the literature, which requires full channel information but not feedback.

Index Terms—User cooperation, video multicast, incremental parity transmission, randomized distributed space time coding.

I. INTRODUCTION

Wireless multicast is an efficient approach to deliver popular live video content to many wireless nodes. However, variations in sender-receiver channel conditions among the receivers make wireless video multicast a challenging problem. Cooperative communication techniques effectively combat such channel quality variations [1]. To enable multiple nodes to cooperate simultaneously, one possible way is to use distributed space-time codes (DSTC) [2]. However, DSTC requires a predetermined and fixed number of relays, and requires tight coordination and synchronization among the relays. To relax these restrictions, Randomized DSTC (R-DSTC) [3] lets each relay transmit a random linear combination of antenna waveforms. This enables all nodes to join in the relaying phase, without requiring strict coordination and synchronization.

A cooperative medium access control (MAC) layer protocol, STiCMAC, designed to allow multiple relays to transmit at the same time using R-DSTC is studied in [4] for an IEEE 802.11 network and in [5] for WiMAX. A joint physical and MAC design for unicast transmission using a randomized cooperative scheme is described in [6]. As stated above, RDSTC is especially attractive for multicast since the nodes that receive the packets can act as relays and transmit simultaneously, without the need for relay selection and scheduling.

To compensate for packet loss during transmission, packet level forward error correction (FEC) [7], [8], [9] and Automatic Repeat reQuest (ARQ) [10] are widely used. In our previous work [11] and [12], packet level FEC using Reed-Solomon codes was employed in conjunction with cooperative transmission.

Cooperative wireless video multicast has been studied by different groups. A cooperative wireless broadcast system for scalable video using a conventional space-time code without considering error control is proposed in [13]. In [14], a joint optimization problem for rate control and relay selection of cooperative wireless video streaming is studied. The impact of cooperative relaying on uplink multi-user (MU) wireless video transmissions is investigated in [15]. The authors simplify this problem as a multi-user Markov decision process and then propose a pricing-based distributed resource allocation algorithm. Neither of these prior works explicitly consider the transmission of error-correction parity packets within the cooperative transmission framework.

In [11], we first proposed to use randomized cooperation for video multicast. In this system, the source (access point, or AP) transmits a video packet, and then all nodes receiving the packet forward this packet simultaneously using R-DSTC. To combat packet losses, the source sends both the original video packets (called source packets) as well as parity packets to enable recovery of lost source packets at the receivers. Each packet goes through two hops, first by the sender, and second by the receivers who received the sender transmission. This original system will be referred as multicast-RDSTC.

An improved parity packet transmission scheme for multicast-RDSTC was proposed in [12], named enhanced-multicast-RDSTC. Here, the source packets are transmitted in two hops, first by the AP and second by the relays. Upon the completion of $k$ source packet transmissions, the nodes that can correctly receive all source packets generate parity packets and transmit them using R-DSTC. As more parity packets are transmitted, more relays join in parity packet generation and transmission. Simulations show that enhanced-multicast-RDSTC can yield a significant increase in the sustainable video rate compared to multicast-RDSTC, by sending the parity packets only once, using relays.

In this paper, we propose a more efficient way for implementing cooperative transmission in video multicast using R-DSTC. The AP will first send all source packets without using relays. After transmitting $k$ source packets (each using only one hop), the source will start to generate and transmit parity packets. Nodes which receive all source packets will also join...
in the generation and transmission of the first parity packet. More nodes will join the parity generation and transmission as soon as they receive a total of \( k \) packets (source or parity), and can therefore decode all \( k \) source packets. To inform all users in the system whether additional parity packets are needed, we further propose an efficient feedback mechanism. After transmission of all source packets, and after each additional parity packet, nodes which still have not received at least \( k \) packets will simultaneously send a request for more parity packets using R-DSTC. We assume that through signaling between the source and receivers and among the receivers, the sender can acquire complete channel information. For each possible channel state (quantized channel condition), we pre-determine the optimal transmission rates for the source packets and parity packets, respectively, which maximize the achievable video rates at all receivers. This new scheme is called as CIPT-multicast-RDSTC, where CIPT stands for Cooperative Incremental Parity Transmission.

The CIPT-multicast-RDSTC system described above requires full-channel information and receiver feedback, which may not be feasible in practical systems. We also consider three other variations that require less channel information and/or feedback. Specifically, in lieu of the complete channel information, we also propose to optimize the system operation based on the node count, which is the total number of receiving nodes. The four variations differ in whether they require full channel information or node-count only, and whether they require feedback.

Simulation results show that the CIPT-multicast-RDSTC scheme requiring full channel information and feedback provides significant gains over the previous multicast-RDSTC scheme, increasing the achievable video rate by 27% on average over a range of node counts from 16 to 80. Compared to the enhanced-multicast-RDSTC, which also requires full channel information feedback, it increases the rate by 15%. Even with only node count information and without feedback, the new scheme provides about 17% improvement over the previous multicast-RDSTC scheme that requires full channel information but no feedback, and provides a gain of 30% over the multicast-RDSTC scheme that requires only the node count information.

The CIPT-multicast-RDSTC system was first described in [16]. This journal version provides additional information regarding the implementation of the feedback and its impact on the achievable video rate. The earlier work in [16] presented three variations of the CIPT system, full-channel with feedback, node-count with feedback, and node-count without feedback. The journal version additionally considers a system requiring full-channel information without feedback, to enable fair comparison with the original RDSTC-multicast system, which requires full channel information without feedback. Furthermore, this version includes additional simulation results that evaluate the impact of the FEC block size on the system performance. Additional figures are added to help the exposition of the proposed system.

The proposed CIPT-multicast-RDSTC system is related to the use of network coding [17] for user cooperation in multicast services, where nodes receiving original source packets generate and send random linear combination of received packets to others. Applications of such schemes for wireless video multicast have been considered by several research groups recently. A comprehensive review of these works can be found in [18]. A structured network coding scheme was introduced in [19] to achieve cooperative peer-to-peer repair for wireless wide area network (WWAN) video broadcast application. However, in these prior works, only one relay can send the coded packets at a time, and relay transmissions must be carefully scheduled to avoid interference and collision. Through R-DSTC, relays in the proposed system can transmit the coded packets simultaneously, and at a higher rate. Because all relays must send the same parity packet at the same time, we use a fixed Reed-Solomon code rather than a random linear code.

This paper is organized as follows. In Section II, our previous work on cooperative video multicast scheme is reviewed. We introduce our proposed system in Section III. The simulation setup and results are presented in Section IV. We conclude the paper in Section V.

II. REVIEW OF PREVIOUS VIDEO MULTICAST SCHEME WITH R-DSTC

This section summarizes our group’s previous work [11][12], where we studied video multicast in an infrastructure-based wireless network. The AP transmits the video to all nodes in a multicast session within its coverage range as shown in Figure 1. We assume that all channels between the receiving nodes and the AP and between the nodes undergo independent slow Rayleigh fading, and the fading level is constant over the duration of single packet transmission. We further assume that a pre-determined set of transmission rates can be achieved using different modulation and channel coding schemes. We refer the readers to [11] for details.

A. Multicast-RDSTC

In the multicast-RDSTC scheme, the video packets are broken into groups of \( k \) packets. For each group of \( k \) source packets, the AP generates \( m \) parity packets, and transmits these packets sequentially at a transmission rate of \( R_1 \) bits/sec. Nodes that receive each packet (either a source packet or a parity packet) correctly, relay this packet simultaneously using R-DSTC with STC dimension \( L \) (which is the number of antennas used for the STC) to other nodes at a transmission rate of \( R_2 \) bits/sec, as illustrated in Figure 2(a). Notice that each packet is transmitted with two hops. In order not to increase the total radiated power over the air, each relay transmits with a power that is equal to the transmission power of the AP divided by the average number of relays for a given node count (the number of users in the multicast session). We assume such information can be predetermined through simulations [11].

Note that there are only a few options for the STC dimension \( L \) in the R-DSTC implementation due to the limited dimensions of practical STC codes. Each \( L \) has a corresponding STC code rate which will affect the effective transmission rate.
Table I

<table>
<thead>
<tr>
<th>Notation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$R_1$</td>
<td>First hop transmission rate for multicast-RDSTC (bits/sec)</td>
</tr>
<tr>
<td>$R_2$</td>
<td>Second hop transmission rate for multicast-RDSTC (bits/sec)</td>
</tr>
<tr>
<td>$R_s$</td>
<td>Source packets transmission rate for CIPT-multicast (bits/sec)</td>
</tr>
<tr>
<td>$R_p$</td>
<td>Parity packets transmission rate for enhanced-RDSTC-multicast</td>
</tr>
<tr>
<td>$L$</td>
<td>Space time code dimension</td>
</tr>
<tr>
<td>$\gamma$</td>
<td>FEC rate</td>
</tr>
<tr>
<td>$k$</td>
<td>Total number of source packets per FEC block</td>
</tr>
<tr>
<td>$m$</td>
<td>Parity packets needed per FEC block</td>
</tr>
<tr>
<td>$N_T$</td>
<td>Total number of nodes</td>
</tr>
<tr>
<td>$R_v$</td>
<td>Achievable video rate (bits/sec)</td>
</tr>
</tbody>
</table>

A full rate can be realized only when $L = 2$ [11]. Therefore, in all results reported here, we use $L = 2$. Optimization of the other parameters is discussed in Sec. II-C.

B. Enhanced approach for multicast-RDSTC

Multicast-RDSTC and enhanced-multicast-RDSTC [12] differ in the way parity packets are sent. For multicast-RDSTC, the AP transmits both the source and parity packets at rate $R_1$, and each packet is then relayed once at rate $R_2$ by nodes that receive the packet.

With the enhanced-multicast-RDSTC scheme, the AP only transmits the source packets. The AP transmits the source packets at a transmission rate of $R_1$ bits/sec and the relays forward these packets using R-DSTC at a rate $R_2$ bits/sec. After the 2-hop transmission of $k$ source packets, the nodes that receive all $k$ source packets generate parity packets and transmit them using R-DSTC at a rate $R_p$ bits/sec. Note that after each parity packet transmission, any node that receives a total of $k$ packets out of all packets transmitted so far, and can therefore decode to obtain the $k$ source packets, becomes a parity relay, and joins the parity packet transmission. Therefore, the number of parity relays increases with time. Figure 2(b) illustrates this scheme. Notice that the source packets go through two hops, whereas the parity packets are sent using one hop only.

Fig. 1. Multicast system layout. The AP is located at the center, with end users randomly placed around the AP within its coverage area, which is a circle of radius $r$.

Fig. 2. Transmission rates and time scheduling for different multicast schemes using RDSTC

C. Optimization Based on Channel Information

In the “full channel” case of multicast-RDSTC, the system requires message exchange between the AP and nodes, and between pairs of nodes, to determine the average channel quality between the AP and each node, and between all nodes, which can be described by a channel quality matrix.
For each particular node placement (with a corresponding channel quality matrix), different candidates of transmission rate combinations \((R_1, R_2)\) are examined. For each candidate \((R_1, R_2)\), the maximum end-to-end PER (averaged over fading) among all nodes can be determined. Based on this a suitable FEC rate \(\gamma\) to ensure that the maximum FEC decoding failure probability among all nodes is less than \(\tau\) can be found. Note that \(\gamma\) is the packet level FEC rate, which can be written as \(\gamma = k/(k + m)\), where \(k\) is total number of source packets per FEC block and \(m\) is parity packets needed per FEC block. Note that compressed video is packet-loss tolerant, our perceptual observations suggest that as long as the FEC decoding failure probability is below a threshold \(\tau = 0.5\%), the loss effect is hardly noticeable, so that the decoded video quality is almost equal to the encoded video quality, which in turn depends only on the video rate[11]. This observation can also be validated by the model in [20].

Given the transmission rates and its corresponding FEC rate, the achievable video rate can be determined. The transmission rate candidate and the corresponding FEC rate that yields the largest video rate constitutes the optimal operating point for this channel quality matrix. We assume that such optimal operating points can be predetermined by simulations for various likely full channel quality matrices, and saved in the AP. The AP periodically collects channel quality information to update its channel quality matrix, and consequently the operating point.

A simplified realization for multicast-RDSTC is also considered, referred as “multicast-RDSTC-node-count”. It does not require the full channel information, but only the node count of the multicast session. To determine the optimal operating point for a given node count, multiple node placements where nodes are uniformly distributed are randomly generated. For each candidate transmission rate combination \((R_1, R_2)\), the worst 5% node placements with worst maximum Packet Error Rates (PER) are not considered. The maximum average PER among all remaining 95% node placements are found, the corresponding \(\gamma\) and hence the video rate based on this PER are then computed. \((R_1, R_2)\) and the corresponding \(\gamma\) are chosen to maximize the video rate. In practice, a table of the system operating parameters \((R_1, R_2, \text{ and } \gamma)\) for different node counts can be pre-computed and stored at the source station.

For the enhanced-RDSTC system, we only considered the version assuming full-channel information. This system needs users to send feedback to help the AP decide when to start transmitting new FEC blocks. This type of feedback information is also needed in the proposed CIPT system, and will be discussed in Section III-B.

III. PROPOSED VIDEO MULTICAST SCHEME WITH INCREMENTAL PARITY PACKETS

A. System Model Overview

In the CIPT-multicast-RDSTC scheme, the source packets are sent only once by the AP at transmission rate \(R_s\) bits/sec and are not relayed. We call the source packets transmission as phase 1. After phase 1 is completed, the parity transmission starts, which is phase 2. If no parity relay are available at the beginning of phase 2 (which will often be the case for systems with low user density), the AP will multicast parity packets by itself.

The parity packets are generated and transmitted incrementally. The process is illustrated in Figure 3. In the beginning of phase 2, only nodes which receive all source packets join the AP to generate and send the first parity packet. Nodes who receive a total of \(k\) packets (out of \(k\) source packets, and the first parity packet) can now recover \(k\) source packets and join the previous parity transmission group to generate and send the second parity packet. As more parity packets are transmitted, more nodes join the parity transmission group. This process continues until all nodes receive \(k\) packets out of all transmitted source and parity packets. Note that we assume there is a short feedback phase after the transmission of each packet, for users who have not received \(k\) packets to indicate that they need to receive more packets. All parity packets will be sent using R-DSTC at transmission rate \(R_p\) bits/sec. The total number of parity packets \(m\), and hence the FEC rate, depends on both the source transmission rate \(R_s\) and the parity transmission rate \(R_p\). As with the previous system, both the AP and the other parity relays transmit at a power that is equal to the AP transmission power in phase 1 divided by the expected number of the parity transmission nodes. Note that this number increases as more parity packets are transmitted. We assume that such information can be estimated from prior simulation studies.

If the average packet size is \(B\), the transmission rate for the source packet is \(R_s\), the transmission time for sending \(k\) source packets is \(T_s = kB/R_s\). Similarly, the transmission time for sending \(m\) parity packets at transmission rate \(R_p\) is \(T_p = mB/R_p\). The video rate \(R_v\) is therefore:

\[
R_v = \frac{\beta k B}{T_s + T_p} = \frac{\beta k R_s R_p}{k R_s + m R_s} = \frac{\beta \gamma R_s R_p}{\gamma R_p + (1 - \gamma) R_s}
\]  

(1)

The symbol \(\beta\) is introduced to account for the fact that only a part of the total sustainable rate is used for sending the video data. Specially, \(\beta\) is the ratio of the data rate used to transmit video data to the total sustainable rate, which includes overhead such as packet headers.

B. Implementation of Channel Information Update and Feedback

There are two types of message exchanges that are needed in the CIPT-multicast-RDSTC system. The first one is needed for updating the channel information, which consists of all signaling needed to deduce the average channel quality in terms of the channel SNR between the source and all the nodes, and between all pairs of nodes. In order for the source station to know the average channel quality among the nodes, the nodes could exchange control signals among themselves to measure the average SNR, and then transmit this information back to the source station. Because the channel information is used to determine the operating parameters for transmitting each FEC block, we envision that such an update to be done at the beginning of every new FEC block or every several FEC blocks, depending on the expected channel dynamics.
Fig. 3. User status during one FEC block (with \(k\) source packets) transmission for CIPT system: (a) The AP transmits all \(k\) source packets; (b) after source packets transmission, several users received all \(k\) source packets; (c) users who receive all \(k\) source packets generate and transmit first parity packet along with AP simultaneously using RDSTC (incremental parity packets transmission started); (d) users who receive a total of \(k\) packets will recover the original \(k\) source packets, and join the previous relay set and the source to generate and simultaneously transmit the second parity packet; (e) as more parity packets are transmitted, more users will be able to correctly recover all source packets and join the parity transmission; (f) once every user receives \(k\) packets, this FEC block is completed.

According to the updated channel quality matrix, the channel is classified into one of several pre-determined channel states, each representing a cluster of channel conditions with similar channel quality matrices. Each state is described by the average channel quality matrix of the cluster. We assume that through pre-simulations, the optimal operating points in terms of \(R_s\), \(R_p\) and possibly the required number of parity packets \(m\) are determined (for no feedback system) and saved for all possible channel states. The information regarding the average number of available relays at each additional parity packet transmission time is also predetermined and saved, to enable power normalization for the parity transmission. Once the AP updates its channel state, it broadcasts the optimal operating point as well as the desired parity transmission powers at different times to all nodes, so that all nodes who can join parity transmission at some time can use the correct transmission rate and power levels.

The second type of message exchange is needed to inform the system when to stop transmitting more parity packets. We assume that by the end of the transmission of every new parity packet, there will be a pre-allocated time slot for users who have not received \(k\) packets to multicast a feedback to indicate that at least one user still needs to receive parity packets. Thanks to the nature of R-DSTC, users who have not received \(k\) packets can multicast the feedback signal simultaneously using only one time slot. The AP and nodes who have received \(k\) packets will listen for this feedback signal at the end of each parity packet transmission, and will keep generating and sending additional packets until no feedback signal is received.

The feedback signal can be any short packet that can be sent over the designated feedback slot. For a typical WiFi environment, for instance, under the 802.11g [21] framework, we could use the PLCP Preamble as our desired feedback message, which will take only \(B_{fb} = 72\text{bits} = 9\text{bytes}\). To guarantee that the feedback packet can be received correctly by nodes everywhere in the multicast session, we use the lowest transmission rate possible, which is \(R_{fb} = 6\text{Mbps}\) in the 802.11g environment. The transmission time for \(m\) feedback packets is \(T_{fb} = mB_{fb}/R_{fb}\). The video rate considering feedback can be derived as:

\[
R_v_{fb} = \frac{\beta k B}{T_s + T_p + T_{fb}} \quad \beta k R_s R_p R_{fb} \quad \frac{\beta k R_s R_p R_{fb}}{k R_p R_{fb} + m R_s R_{fb} + \alpha m R_s R_p}
\]

where \(\alpha = B_{fb}/B\). The rate reduction factor due to feedback
is thus
\[ \omega = \frac{R_{fb}}{R_v} = \frac{(1 - \gamma)p_s + \gamma p_p}{\gamma p_v + (1 - \gamma)(1 + \alpha p_p)p_s} \]  \quad (3)

where \( p_s = R_s/R_{fb} \) and \( p_p = R_p/R_{fb} \).

If we assume the video packet size is \( B = 1400\text{bytes} \) and the feedback packet size \( B_{fb} = 9\text{bytes} \), then \( \alpha = 0.0064 \). For our proposed system, the optimal operating point for most user configurations consists of \( R_s = 24\text{Mbps} \), \( R_p = 36\text{Mbps} \) and \( \gamma \approx 0.5 \). With \( R_{fb} = 6\text{Mbps} \), we have \( p_s = 4 \) and \( p_p = 6 \). With these assumptions, we get \( \omega = 0.9848 \), meaning that the feedback overhead is about 1.5%.

C. Four Variations of the CIPT-multicast System and Optimization of Their Operating Parameters

By switching on and off the above two types of message exchanges, we will have four variations of the CIPT-multicast system. We will discuss optimization of the operating parameters for each variation in this subsection.

<table>
<thead>
<tr>
<th>Algorithm 1: Calculation of ( m ) for one FEC block for given ((R_s, R_p)) and full channel state</th>
</tr>
</thead>
<tbody>
<tr>
<td>Run channel simulation to transmit ( k ) source packets from AP, to determine the number of source packets received for each user</td>
</tr>
<tr>
<td>Form relay set (users who received all ( k ) source packets) ( m = 0 )</td>
</tr>
<tr>
<td>while at least one user needs more packets do</td>
</tr>
<tr>
<td>Run channel simulation to transmit one parity packet by AP and all users in the current relay set using R-DSTC to determine whether the parity packet is received for each remaining user</td>
</tr>
<tr>
<td>( m = m + 1 )</td>
</tr>
<tr>
<td>Update relay set</td>
</tr>
<tr>
<td>return ( m )</td>
</tr>
</tbody>
</table>

1) Full channel information with feedback: This scenario assumes the AP periodically updates the channel information to determine the channel state, as described in Sec. III-B. Furthermore, it assumes that users are able to multicast feedback packets to make the system determine the termination time for each FEC block.

To determine the optimal operating point for each channel state, we go through all possible pairs of feasible \( R_s \) and \( R_p \). For each candidate pair of \( R_s \) and \( R_p \), through channel simulations, we determine the necessary number of parity packets \( m \) for the transmission of each FEC block (using Algorithm 1), so that all nodes receive at least \( k \) packets. We find the average of \( m \) over many FEC blocks. We use this \( m \), in addition to \( R_s \) and \( R_p \), to determine the video rate using Eq.(2). Finally we choose the optimal \( R_s \) and \( R_p \) that maximizes the video rate. This procedure is summarized in Algorithm 2. Details regarding channel simulations both for the direct transmission by AP and for cooperative transmission by AP and relays using R-DSTC can be found in [11].

The optimal \( R_s \) and \( R_p \) for different channel states are stored in a lookup table. Note that \( m \) does not need to be stored in the look up table, as we use feedback to determine the necessary \( m \) for each particular FEC block.

2) Full channel information with no feedback: This system is similar to the previous system, except that we do not rely on feedback to decide when to stop sending the parity packets. Instead, for each channel state, we predetermine the number of parity packets to be sent for each candidate transmission rate pair \((R_s, R_p)\), so that the maximum FEC decoding failure rate among all users is below a preset threshold \( \tau \). The algorithm used to determine \( m \) for given \((R_s, R_p)\), and the optimal \((R_s^*, R_p^*)\) is described in Algorithm 3. As with the previous multicast-RDSTC scheme for determining the necessary parity packet number, we set \( \tau = 0.5\% \), because this low FEC decoding failure rate leads to negligible perceptual distortion due to packet loss [11].

Unlike the previous scenario, this time \( m \) for different channel states will also be stored in a lookup table to let the AP and all the users know when to terminate the transmission of the current FEC block without utilizing feedback.

3) Node-Count information with feedback: This case assumes that the AP only knows the node count. It still requires feedback from all nodes to determine when the parity transmission should be terminated. As with the node-count version of the multicast-RDSTC system [11], for each possible node count, we generate multiple node placements using uniform distribution of nodes in the coverage area. For each candidate pairs \((R_s, R_p)\), we remove the worst 5% of node placements. Specifically, as shown in Algorithm 4, we find the average parity packet number \( m^* \) needed for each node placement over many FEC blocks. We remove 5% of node placements with largest \( m \). Then we find the maximum parity packet number \( m^* \) needed among all remaining node placements. We compute the video rate corresponding to this \( m^* \) and the rate pair \((R_s, R_p)\). Finally we choose the rate pair \((R_s^*, R_p^*)\) that achieves the highest video rate as the optimal operating point for this node count. The optimal \( R_s^* \) and \( R_p^* \) for different node counts will be pre-computed and stored in a look-up table at the AP.

4) Node-Count information with no feedback: As shown in Algorithm 5, we find the maximum parity packet number \( m^* \) needed for each node placement over many FEC blocks so that the FEC decoding failure rate is below \( \tau = 0.5\% \). Then as in the previous case, we remove the 5% of node placements with the largest \( m \), and find the maximum parity packet number \( m^* \) needed among all remaining node placements. We determine the video rate based on \((R_s^*, R_p^*)\) and \( m^* \) for each node count. We record not only the optimal \((R_s^*, R_p^*)\) but also the corresponding maximum number of parity packets \( m^* \) for the chosen \((R_s^*, R_p^*)\). All this information will be precomputed and stored in a look-up table. As is apparent, this system will be the easiest one to implement.

We have relied on using channel simulations and numerical search to find the optimal operating points of the proposed system variations. Theoretical derivation of such parameters is extremely challenging, especially because the packet loss rates seen by different remaining users in the cooperative transmission of each additional parity packet depend on which users are in the current relay set, and the relay set changes for
interesting trends of how these operating parameters change each new parity packet. Nevertheless, we have found some interesting trends of how these operating parameters change with the node count, which will be discussed in the next section.

**Algorithm 2:** Determine optimal $R_s, R_p$ for the full channel with feedback system

<table>
<thead>
<tr>
<th>for each channel state do</th>
</tr>
</thead>
<tbody>
<tr>
<td>for each candidate pair $(R_s, R_p)$ do</td>
</tr>
<tr>
<td>for each FEC block do</td>
</tr>
<tr>
<td>- Determine $m(R_s, R_p)$ using Algorithm 1</td>
</tr>
<tr>
<td>- Determine $m^<em>$ so that the FEC decoding failure rate among all FEC blocks among all users is below $\tau$ ($m^</em>$ equals the $(\tau \cdot L + 1)$-th largest $m$, where $L$ is the number of FEC blocks transmitted)</td>
</tr>
<tr>
<td>- Determine the video rate using Eq. 1 and $(m^*, R_s, R_p)$</td>
</tr>
<tr>
<td>- Determine $(R_s^<em>, R_p^</em>)$ that maximizes the video rate</td>
</tr>
<tr>
<td>- Record $(R_s^<em>, R_p^</em>)$ for each channel state in a look-up table</td>
</tr>
</tbody>
</table>

**Algorithm 3:** Determine optimal $R_s, R_p, m$ for the full channel without feedback system

<table>
<thead>
<tr>
<th>for each channel state do</th>
</tr>
</thead>
<tbody>
<tr>
<td>for each candidate pair $(R_s, R_p)$ do</td>
</tr>
<tr>
<td>for each FEC block do</td>
</tr>
<tr>
<td>- Determine $m(R_s, R_p)$ using Algorithm 1</td>
</tr>
<tr>
<td>- Determine $m^*$ so that the FEC decoding failure rate among all FEC blocks among all users is below $\tau$</td>
</tr>
<tr>
<td>- Determine the video rate using Eq. 1 and $(m^*, R_s, R_p)$</td>
</tr>
<tr>
<td>- Determine $(R_s^<em>, R_p^</em>)$ that maximizes average video rate</td>
</tr>
<tr>
<td>- Record $(R_s^<em>, R_p^</em>)$ and $m^*$ for each channel state in a look-up table</td>
</tr>
</tbody>
</table>

**Algorithm 4:** Determine optimal $R_s, R_p$ for the node count with feedback system

<table>
<thead>
<tr>
<th>for each node count do</th>
</tr>
</thead>
<tbody>
<tr>
<td>for each candidate pair $(R_s, R_p)$ do</td>
</tr>
<tr>
<td>for each node placement (and corresponding channel state) with this node count do</td>
</tr>
<tr>
<td>for each FEC block do</td>
</tr>
<tr>
<td>- Determine $m(R_s, R_p)$ using Algorithm 1</td>
</tr>
<tr>
<td>- Determine $\hat{m}$ as the average $m$ over all FEC blocks</td>
</tr>
<tr>
<td>- Determine 5% node placements with largest $\hat{m}$</td>
</tr>
<tr>
<td>- Determine $m^*$ as the maximum $\hat{m}$ among remaining node placements</td>
</tr>
<tr>
<td>- Determine the video rate using Eq. 1 and $(m^*, R_s, R_p)$</td>
</tr>
<tr>
<td>- Determine $(R_s^<em>, R_p^</em>)$ that maximizes the video rate</td>
</tr>
<tr>
<td>- Record $(R_s^<em>, R_p^</em>)$ for the node count</td>
</tr>
</tbody>
</table>

**Algorithm 5:** Determine optimal $R_s, R_p, m$ for the node count without feedback system

<table>
<thead>
<tr>
<th>for each node count do</th>
</tr>
</thead>
<tbody>
<tr>
<td>for each candidate pair $(R_s, R_p)$ do</td>
</tr>
<tr>
<td>for each node placement with this node count do</td>
</tr>
<tr>
<td>for each FEC block do</td>
</tr>
<tr>
<td>- Determine $m(R_s, R_p)$ using Algorithm 1</td>
</tr>
<tr>
<td>- Determine $\hat{m}$ so that the FEC decoding failure rate among all FEC blocks among all users is below $\tau$</td>
</tr>
<tr>
<td>- Determine 5% node placements with largest $\hat{m}$</td>
</tr>
<tr>
<td>- Determine $m^*$ as the maximum $\hat{m}$ among remaining node placements</td>
</tr>
<tr>
<td>- Determine the video rate using Eq. 1 and $(m^*, R_s, R_p)$</td>
</tr>
<tr>
<td>- Determine $(R_s^<em>, R_p^</em>)$ that maximizes the video rate</td>
</tr>
<tr>
<td>- Record $(R_s^<em>, R_p^</em>)$ and $m^*$ for the node count</td>
</tr>
</tbody>
</table>

IV. Simulation Results

We have simulated video multicast using different versions of the CPT transmission scheme, as well as direct transmission, multicast-RDSTC [11] and enhanced-RDSTC-multicast [12]. For comparison we also compare with a multicast system that only uses direct transmission from the AP. In our simulations, we generate 300 random node placements for each node count, and report the average performance over these 300 node placements for each node count. There are five node counts considered, which are 16, 32, 48, 64, and 80 nodes. We assume that each FEC block has $k$ source packets. In most of our results, the FEC block size is set at $k = 64$. We have also examined other possible values of $k$, including 4, 8, 16, and 32, to evaluate the impact of $k$ on the system performance.

For each particular node placement for a given node count, we simulate the transmission of 20,000 FEC blocks, with independent fading realizations among these blocks, and report the average performance over these blocks. We keep the fading level constant within each packet and independent among all packets. Through channel simulation, we determine whether a source packet send by the AP only using a given source transmission rate is received by each node, and whether a parity packet sent by the AP and relay nodes using RDSTC at an assumed parity transmission rate is received by each remaining node. The simulation procedure can be found in [11].

For the systems assuming full channel information (including CPT-multicast and multicast-RDSTC), to find the optimal operating point (in terms of transmission rates for different transmission phases and corresponding FEC rate) for a particular node placement, we treat each node placement as a separate channel state, and use the method described in Sec.III-C1 to determine the optimal operating point for each state. For the systems that assume only the knowledge of the node count (including CPT node count w/feedback, CPT node count w/o feedback, and RDSTC-multicast node count),
for a given node count, among the 300 node placements, we identify 5% of those (i.e., 15 node placements) that have the worst end-to-end packet error rate (PER) (for CIPT-multicast systems, it means largest \( m \)), and determine the optimal operating point for the remaining 95% node placements as described in Section II-C and III-C.

For the baseline direct-multicast transmission system, the source station transmits the packets at the base rate of the underlying network (e.g., 6 Mbps for IEEE 802.11g) with a packet level FEC rate \( \gamma \) that is chosen based on maximum PER among all users and all node placements such that the FEC decoding failure probability for every user is smaller than \( \tau \).

Figure 4 compares the achievable video rates by different transmission schemes. For all cooperative schemes, the achievable video rate increases with the node count. This is because, with more users in a multicast session, thanks to the nature of cooperative communication, we can have more nodes available to help in the parity packets transmission phase, which reduces the number of parity packets required.

Figure 4 shows that CIPT scheme outperforms multicast-RDSTC and enhanced-RDSTC-multicast in every comparable situation. For example, CIPT full channel w/o feedback can increase the video rate by 22% over the multicast-RDSTC full channel, both requiring full channel information, but no feedback. When both are using only node count information, CIPT node count w/o feedback improves over RDSTC node count by 30%. Furthermore, CIPT full channel w/feedback outperforms enhanced RDSTC full channel w/feedback by 15%.

Among the CIPT systems, compared to CIPT with full channel information and feedback (denoted by CIPT full channel w/feedback), using node count information only but retaining feedback (denoted by CIPT node count w/feedback) leads to small degradation in video rates when the node count is small. As the node count increases, the difference becomes negligible. This is as expected; with more nodes, which are uniformly distributed, the optimal operating point depends less on the actual node placement. When the system still assumes full channel information, but does not require feedback (CIPT full channel w/o feedback), the performance degradation is larger, and remains non-negligible even for larger node counts. This is because feedback enables the system to provide just enough parity packets for each FEC block. Without feedback, the system has to choose \( m \) in a conservative fashion to guarantee that all users receive at least \( k \) packets within each transmission block. However, as demonstrated in Figure 4, the degradation due to the removal of feedback is relatively small, with a video rate reduction of 4%, slightly more with fewer nodes. Comparison of “CIPT full channel w/o feedback” and “CIPT node count w/feedback” reveals that the feedback information is more important than the full channel information. When we remove the requirement for both the full channel information and feedback (i.e. “CIPT node count w/o feedback”), the system performance is decreased further, with a video rate decrease of 5%-12% compared to the full-channel with feedback case. However, such performance degradation may be well justified in practice, because this simplified system does not require any feedback or message exchange among the users and the source station. Even with this most simplified CIPT system, the video rate increases by 17% over the previous multicast-RDSTC system using full channel information.

Recall that when using node count information to substitute for the full channel information, we ignore node placements with the worst 5% performance when determining the optimal operating point for each node count. Similarly, when foregoing the feedback information, we choose the number of parity packets to be sent so that each user will see at most \( k \) packets within each transmission block. Figure 5 shows that for the CIPT node count w/o feedback system, among all nodes in all node placements considered (including worst 5%), the percentage of users that can not recover all source packets is less than only 0.43%.

Figure 6(a) shows optimal transmission rates for the three multicast systems that use R-DSTC. Figure 6(b) shows the corresponding FEC overhead \( m/k \) needed to recover packet loss. The systems considered in this figure are all with full channel information. CIPT-multicast and enhanced-RDSTC are using feedback. We can see that the transmission rates
for first hop source packets $R_1$ by multicast-RDSTC and enhanced-RDSTC are substantially higher than $R_s$, the transmission rate for source packets by CIPT-multicast. Since the former two systems use relays to help with source packet transmission, the transmission rate of first hop can be higher. Part of the gain of enhanced-RDSTC over multicast-RDSTC is due to a much higher transmission rate for parity packets $R_p$ than that used for the second hop transmission for the source packet $R_2$. We believe this is because, typically there are more nodes available for simultaneously sending the parity packets. For the proposed CIPT-multicast system, since source packets and parity packets are transmitted only once, to make sure sufficient number of users receive these, transmission rates for both source packets $R_s$ and parity packets $R_p$ are significantly lower. However, even though $R_s$ and $R_p$ are lower, since both source and parity packets are transmitted only once, and the total number of parity packets needed per FEC block $m$ is generally less than the number of source packets, the overall achievable video rate is still significantly higher than the multicast-RDSTC and enhanced-RDSTC-multicast, as illustrated in Figure 4. Generally, there is a trade-off between the transmission rates and the number of parity (and correspondingly FEC overhead) needed. Through our optimization study for the CIPT-multicast system with full channel information, for the low-node count case, it is better to use lower transmission rates (with correspondingly lower FEC overhead); but for high-node count case, it is better to use higher transmission rates (with correspondingly higher FEC overhead).

Figure 7(a) shows the optimal transmission rate for all four proposed variations of CIPT-multicast system. Figure 7(b) shows the corresponding FEC overhead $m/k$ needed. Note that for the full channel information schemes, the optimal transmission rates showed in the figure is averaged like in Figure 6. As the figure shows, for the no feedback schemes, the optimal transmission rate for source packets $R_s$ is low especially when $N_T$ is larger. However, the optimal transmission rate for parity packets $R_p$ is more similar to the schemes with feedback. For the FEC overhead $m/k$, figure 7(b) shows that for two schemes considering feedback, they have similar number of $m$ needed. The two curves converged when $N_T = 80$. It is also true for the two schemes without feedback. Since the FEC overhead $m/k$ is predetermined using maximum $m$ for the schemes without feedback, the optimal transmission $R_s$ is chosen to be lower than other two schemes to avoid higher $m$. Under the same feedback assumption, CIPT-multicast with full channel information need lower number of parity packets $m$ for the optimal transmission rates. It is because the optimal source transmission rate for the node-count system is higher, which requires more parity packets to protect.

Figure 8 illustrates how the percentage of users who can decode the entire FEC block increases as more parity packets are transmitted, and the influence of transmission rates and node counts on this percentage. As the upper figure shows, when the transmission rate reduces, the percentage increases faster. This is as expected, as lower transmission rates enable more users to receive $k$ packets earlier. The figure further shows that, when the transmission rates ($R_s$ and $R_p$) are the same, the curves corresponding to different node counts (curves with the same color) cluster together. This is an advantage for the practical implementation of the proposed system. Recall that the system needs to know the number of relays participating in simultaneous transmission using R-DSTC, to normalize the transmission power. We could predetermine, through simulations, such a curve for each optimal transmission rate combination, which is the same regardless of the number of nodes, and use these pre-determined curves for power normalization during real time transmissions. It is also possible to determine simple mathematical models for such curves, whose parameters are only dependent on the transmission rates, to further simplify the system implementation.

The lower-left small figure shows that the source packet
transmission rate $R_s$ affects the initial number of user who receive the entire FEC block after the source completes its source packet transmission. As the figure shows, at the very beginning, green and black curves (which have the same $R_s$) are overlapping with each other, and so are red and blue curves (which have the same $R_s$). The lower-right small figure shows the impact of parity packets transmission rate $R_p$ on the number of parity packets needed for all users to receive at least $k$ packets. Basically, a larger $R_p$ requires more parity packets. With the same transmission rates combination ($R_s$, $R_p$), when the node count is larger, the system converges faster (i.e. requires fewer parity packets).

Figure 9 shows the same type of curves for different node counts, for the case when the system uses the optimal transmission rates for the corresponding node counts. Because the optimal transmission rates for node counts $N_T \geq 32$ are similar (cf. Figure 6), these curves are close to each other. The system for $N_T = 16$ converges faster (i.e. requires fewer parity packets) because it uses lower transmission rates. However, because it uses substantially lower transmission rates, the total achievable video rate is lower than systems with larger node counts.

We also explore performance improvement in terms of video quality over previous systems. Figure 10 shows the corresponding PSNR of video sequences Harbour and Soccer coded by a H.264/AVC compliant encoder x264[22] at the maximum achievable video rate for different practical multicast systems. For the two CIPT-multicast systems, we assume there is only node count information available with feedback mechanism on and off. Similarly, for multicast-RDSTC system, we also assume there is only node count information available. For the sequence Harbour, CIPT-multicast node count without feedback has about 1dB gain over multicast-RDSTC when $N_T$ is large. When $N_T$ is small, the gain is greater, up to 1.36 dB. For the sequence Soccer, the gain is between 0.94 to 1.24 dB.

We would like to note that generally a 1dB gain in PSNR is quite noticeable in perceptual quality. It is worth noting that the gain between two CIPT-multicast systems is also considerable when $N_T$ is small. CIPT-multicast node count system with feedback gives about 0.42 dB gain for sequence Harbour and 0.44 dB gain for sequence Soccer over CIPT-multicast node.
with lower node count. This is demonstrated in Figure 11(b), which shows that in general the optimal source and parity transmission rates both decrease with $k$ especially when $N_T$ is large and $k \geq 8$. On the other hand, with a larger $k$, the FEC is more effective in that it requires fewer parity overhead (defined by the ratio $m/k$). This is demonstrated in Figure 12, which shows that a higher overhead is needed for a system with smaller $k$. Because the overall video rate increases with transmission rates but decreases with the overhead rates, some intermediate $k$ is optimal. Figure 12 further demonstrates that under the optimal transmission rates, the necessary overhead rate, $m/k$, is smaller than 1.5 for different $k$ and different $N_T$. For $k > 32$, the ratio $m/k$ is below 1.

![Fig. 12. FEC overhead $m/k$ needed for different FEC block size and different node count. The same color indicates the same node count. Each group of bars have the same FEC block size.](image)

**V. Conclusion**

In this paper, we have proposed an innovative wireless video multicast system which features cooperative incremental parity packet transmission using R-DSTC. Both source packets and parity packets are transmitted using only one hop. Users who receive $k$ packets will generate parity packets and join parity packet transmissions using R-DSTC. In the most ideal, but also the most complex implementation, the system periodically updates the full channel information through message exchange among all nodes and uses feedback from users to determine whether to continue sending parity packets. We have optimized the transmission rates for both the source packets and parity packets for each possible channel state corresponding to the full channel information. Three suboptimal but more practical schemes have also been investigated. The first scheme does not require the full channel information but requires feedback, the second requires the full channel information but not the feedback, and the third requires neither full channel information nor the feedback. We have shown that all different variations of the proposed CIPT scheme provide performances quite close to the ideal case with full channel information and feedback, and achieve substantially higher video rates over the multicast-RDSTC scheme in which both source and parity packets go through two-hop transmission. The CIPT system also provides
significant gain over the enhanced-multicast-RDSTC system, which requires feedback. Our simulation results further have shown that the feedback information is more important than the full channel information in maintaining a high achievable rate.

To further improve the CIPT-multicast-RDSTC scheme, one possibility is to discontinue relays from parity packet transmissions after they have transmitted a certain number of times and hence may not be able to help additional surrounding nodes. By removing them from parity transmission, the remaining parity relays can transmit at higher power. Another avenue is to dynamically increase the parity transmission rate as more parity packets are sent. This is plausible because, as more relays join parity transmission, higher transmission rates are sustainable. Yet another research direction is to adapt the proposed scheme to layered coded video, as done in [11]. For example, we can choose the parity packet number for the base layer video so that all users can recover all source packets with a high probability, and choose the parity packet number for the enhancement layer so that only a certain percentage of users can recover all source packets.

REFERENCES