



Efficient Error Estimating Coding Using OTP Algorithm & GDS Scheme

Y.Naveen

M. Tech Student, SE

Kakatiya Institute of Technology & Science (KITS),
Warangal, India

G.Sridhar

Asst. Professor, CSE

Kakatiya Institute of Technology & Science (KITS),
Warangal, India

Abstract— *Motivated by recent emerging systems that can leverage partially correct packets in wireless networks, this paper proposes the novel concept of error estimating coding (EEC). Without correcting the errors in the packet, EEC enables the receiver of the packet to estimate the packet's bit error rate, which is perhaps the most important meta-information of a partially correct packet. Our EEC design provides provable estimation quality with rather low redundancy and computational overhead. To demonstrate the utility of EEC, we exploit and implement EEC in two wireless network applications, Wi-Fi rate adaptation and real-time video streaming. Our real-world experiments show that these applications can significantly benefit from EEC.*

Keywords— *BRR, Bit Error Rate, Error Estimation Coding (EEC), Mobile Networks*

I. INTRODUCTION

Error correcting coding has long been playing a key role in serving the performance and reliability needs in wireless networks. Over the years, researchers have proposed numerous interesting error correcting codes. The traditional philosophy behind error correction is that the application/network can or should only use/relay completely correct packets. Recent advances in wireless networking, however, have invalidated this assumption. In particular, many designs can now use a packet that is partially correct (i.e., some bits are correct, but others are corrupted). Such a *partial packet* can be useful when the following situations apply.

- The destination may be able to obtain incremental redundancy from the source to recover the partial packet (i.e., incremental redundancy ARQ).
- The destination may collect and combine multiple partial packets to obtain a correct copy.
- The packet has *forward error correction* (i.e., pre-encoded with error correcting codes) and thus can potentially fully recover the errors. For example, forward error correction is often used in real-time video streaming to tolerate errors in wireless networks.
- The application may be able to directly use partial packets to some extent. For example, for image or video packets, a partially correct packet can still carry some useful information.

These designs raise the natural question of whether there is any benefit looking beyond error correcting codes. This paper takes the first step in answering this question by proposing the novel concept of *error estimating coding* (EEC). Without actually correcting the errors in the packet, EEC enables the receiver to estimate the *fraction* of corrupted bits in the packet, which is perhaps the most important meta-information of a partial packet. We call such a fraction the packet's *bit error rate*, or *BER*. Here the receiver of the packet may or may not be the packet's final destination. In particular, it can be a wireless router that is *oblivious* to how the application will eventually use/recover the partial packet. The utility of EEC depends on two key questions. *Feasibility*: Is it possible to construct highly efficient EEC? In particular, EEC's redundancy and computational overhead must be substantially smaller than error correcting codes, since otherwise one should just directly use error correcting codes to correct the errors. *Applications*: Does the BER meta-information provided by EEC significantly benefit upper-layer applications? Affirmative answers to these two questions would imply that EEC indeed achieves a new and interesting tradeoff point, on the spectrum between overhead and functionality. Such tradeoff point was not previously available with error correcting coding. *Efficient EEC—Feasibility*: This paper provides affirmative answers to both questions above, thus confirming the utility of the novel concept of EEC. First, for feasibility, our EEC design only needs to add extra bits to the packet (being the number of data bits) to estimate BER. As concrete numerical examples in the two EEC applications that we implement (described later), the relative redundancy added to each packet is only about 2%. In fact, in cases where we only need to estimate whether the BER exceeds a certain threshold; the redundancy added by EEC is only 4 B (as in one of our two applications). Such a small redundancy makes it possible to even view EEC as generalized CRC. Namely, CRC tells us whether the BER exceeds 0, while EEC can tell whether the BER exceeds any given threshold. We also trivially show that error correcting codes, in order to correct the errors, would require much higher redundancy.

II. PREVIOUS WORK

In wireless relay networks (WRNs) signals are transmitted from one terminal to another through a number of relays. The main advantage of doing so is a reduced signal transmit power. Design of efficient relaying protocols and distributed coding schemes in WRNs has recently attracted a lot of attention. Some cooperative or distributed coding schemes have

been proposed to explore the cooperative spatial diversity and cooperative coding gains in WRNs. However most of the existing distributed coding schemes was based on the amplify and forward (AAF) and decode and forward (DAF) relaying protocols, which suffer from a disadvantage of either noise amplification or error propagation. It is therefore very important to develop a simple but robust relaying protocol. In this paper, we propose a simple adaptive relaying protocol (ARP), which takes advantages of both DAF and AAF protocols and minimize their disadvantages. For the proposed scheme, all relays are included into one of two relay groups, which we called a DAF relay group, and an AAF relay group. All relays, which can correctly decode the signals transmitted from the source, are included in the DAF relay group and the rest of relays, which fail to make a correct decoding, are included in the AAF relay group. All the relays in the AAF relay group amplify the received signals from the source and forward it to the destination, while the relays in the DAF relay group decode the received signals, re-encode and forward them to the destination. The processing at the destination for the ARP is the same as for the AAF and DAF. All signals received at the destination, forwarded from the relays in both the AAF and DAF groups are combined together into one signal.

The performance of the proposed ARP is analyzed and compared with other relaying protocols. It is shown that the proposed ARP scheme considerably outperforms the AAF scheme and circumvents the error propagation due to the imperfect decoding at relays in a DAF protocol, thus considerably outperforming both the AAF and the DAF protocols. This performance gain grows as the number of relays increases and it approaches the perfect DTC at the high signal to noise ratios (SNR). The above analytical results are also validated by simulation results. In a practical system, in order to determine whether a relay can decode correctly or not, some CRC bits can be appended to each information block (frame). After decoding each frame, the relay can examine the CRC checks to determine if the received signals are decoded correctly or not. One important feature of the proposed ARP scheme in a practical application is that the relay can automatically adapt to the channel quality by simply switching between the AAF and the DAF without any need for the channel state information (CSI) to be fed back from the destination to the relays or the source. This feature is very important in practical relay networks, especially in a multi-hop large network, in which the feedback of CSI for adaptation is very expensive. Another important feature is that the processing at relays and destination for the ARP scheme is the same as for the AAF and DAF and it does not add any complexity to the system performance. [1]

The demand for wireless bandwidth in indoor environments continues to increase with the rapid integration of Wi-Fi radios in every day consumer electronic devices such as lap-tops, cell phones, cameras, and audio/video equipment. This increased density of Wi-Fi transmitters has exacerbated contention for the wireless medium and reduced overall throughput. For example, the authors observed that more than 70% of the frames in a conference had at least one contender. One effective approach to improving throughput is to reduce interference among radios, thus allowing them to transmit concurrently.

Directional antennas have the potential to provide the necessary interference reduction by spatially containing transmissions. For example, commercial, off the shelf directional antennas can provide spatial isolation of up to 20 dB by containing the signal within a sector of 10, i.e., the signal outside the sector is at least 20 dB weaker than the signal within the sector. This high degree of spatial isolation can support concurrent transmissions between pairs of radios even when they are located close to one another provided that the antennas can be oriented correctly.

Traditionally, directional antennas have mostly been used in outdoor environments where there is a direct line-of-sight (LOS) between the two endpoints. In such applications, orienting directional antennas is relatively simple because the LOS antenna orientation usually maximizes the signal strength at the receiver and is the only configuration that works well. In an indoor environment, the presence of rich scattering and multipath effects results in non-LOS antenna orientations that provide comparable signal strength at the receiver. In fact, conventional wisdom has been that directional antennas will not be effective indoors because of this issue. Although the existence of these alternate good orientations complicates the configuration of the system, it also creates an opportunity for reducing interference between transmissions. For example, while the LOS orientation of a transmitter may cause interference at some location, an alternate configuration that uses a path may steer the signal such that it reaches the intended recipient without causing interference. However, to achieve the gains offered by these alternate configurations requires significant, explicit coordination between the transmitters in the network. For example, coordination is necessary for transmitters to measure the interference created in different antenna configurations. In addition, since the presence of directional antennas exacerbates the presence of hidden and exposed terminals in the network, coordination is also needed to address MAC related problems. The focus of our work is on the use of directional antennas to increase spatial reuse. Unlike omnidirectional antennas that have a uniform gain in each direction, directional antennas have a different antenna gain in each direction. As a result the signal level at a receiver can be increased or decreased simply by rotating the orientation of the directional antenna. In the directional mode, the antennas we use have the ability to increase or decrease the signal strength at a receiver node by up to 20 dB. Received signal strength indoors can be further affected by the presence of strong RF like metal cabinets, walls, and doors, resulting in multiple paths that add up constructively or destructively at a receiver. In addition to directionality, our design also relies on the property of many Wi-Fi radios, called the capture effect. If a card supports capture, it can successfully receive a packet transmission in the presence of concurrent transmissions, as long as the desired transmission has a sufficiently higher signal strength. The signal strength difference required is usually in the range of 20-25 dB for 54 Mbps. The majority of this difference, 10-20 dB, can be provided by appropriately orienting the antennas. The rest of the difference can be achieved from the fact that many receivers are closer to their senders than to the interferers. As a result, with careful configuration of directional antennas, we can often ensure that the intended receivers can "capture" their packets, despite the presence of interfering transmissions. [2]

Rate adaptation protocols adjust the modulation rate according to the quality of the channel. When there is mobility of the sender, receiver, or scatterers within the environment, the channel characteristics change, thereby inducing fluctuations of the channel quality, i.e., channel fading. Depending on the degree of such fluctuations, the previously appropriate rate could become under selected if the channel state has improved or over selected if the channel state has become worse. The inability to accurately choose the modulation rate for the current channel condition leads to loss or unnecessarily long packet transmission times, and hence, inefficient use of the channel.

Rate adaptation protocols address channel fading in one of two ways. In loss-triggered rate adaptation, the transmitter interprets channel state based upon timeouts (failed delivery) or receipt of acknowledgments (successful delivery) following the transmission of data packets. Loss-triggered protocols use this delivery result of multiple packets to determine the appropriate modulation rate, see for example. These protocols require limited MAC and PHY interaction and are widely implemented and evaluated in indoor and outdoor settings. In SNR-triggered rate adaptation, the receiver uses the signal-to-noise ratio to determine the modulation rate and informs the transmitter via the four-way handshake. These protocols have not been implemented previously due to the closed and inflexible MAC and PHY of legacy systems.

In particular, we make the following four contributions. First, we design a cross-layer rate adaptation framework and implement five key mechanisms used by rate adaptation protocols out of which, three are used by loss-triggered protocols and two by SNR-triggered protocols. We are the first to implement SNR-triggered protocols on hardware at MAC time-scales comparable to commercial systems. In in-lab and urban outdoor environments, we evaluate rate adaptation protocols by measuring the success or failure of the protocols' selected rate as compared to the ideal rate. We determine the ideal rate via exhaustive experimental search by replaying channel conditions through multiple rate adaptation mechanisms and experimentally identifying the rate decisions that maximize throughput. In this way, we characterize the multirate mechanisms' inaccurate rate decisions to reveal the origins of poor throughput performance. In contrast, prior work neither compared protocols' rate selection with optimal rate selection nor evaluated rate adaptation decisions on a packet-by-packet basis.

Second, we evaluate rate adaptation accuracy on diverse channel operating conditions including fast-fading, multipath, and interference. We find that as coherence time decreases (fast-fading), both loss-based and SNR-based mechanisms have low throughput. However, we show via per-packet evaluation that this poor performance is due to opposite rate selection inaccuracies: Loss-triggered mechanisms under select when they require consecutive successful packets to increase their transmission rate, as this occurs with low probability in fast-fading environments. In contrast, SNR-triggered protocols over select with a fast-fading channel due to sensitivity to coherence time. Yet, we show that when SNR protocols are trained according to the environment's coherence time, significant throughput gains can be achieved. Further, we show that the need for such training increases with the presence of multipath, an effect we observe to be strongly present within the downtown scenario but not within the residential urban environment. Third, with controlled in-lab experiments, we investigate rate adaptation accuracy with heterogeneous links (links of differing average quality), as commonly measured in outdoor environments. We show that a protocol designed to overcome the misinterpretation of collision-based losses and fading-based losses with out-of-range senders (the hidden terminal scenario) is effective (i.e., high aggregate throughput and equal sharing) when the competing links are statistically equal in quality. However, we find that the protocol has a severe throughput sharing imbalance whenever even slight differences in average link quality exist between competing transmitters. We show that this is due to the slight difference in channel quality driving the system to a state in which only one transmitter uses the four-way handshake significantly more often, thereby giving it increased protection from hidden terminal collisions. With higher link heterogeneity between competing transmitters, the physical layer capture effect occurs in which the stronger link is able to successfully transmit packets to the receiver even with simultaneous transmissions from a weaker transmitter. We present the first evaluation of rate adaptation performance coupled with capture and find that their joint interaction can cause significant unnecessary reductions in modulation rate. [3] Wireless LANs based on 802.11 are used almost everywhere, from airports to zoos and in urban, suburban and rural areas. Modern wireless NICs provide a large and growing range of physical layer configurations to obtain good performance across this range of environments. With 802.11n, the latest version of the standard that ships on most laptops, combinations of modulation, coding and spatial streams offer rates from 6 Mbps to 600 Mbps. Other important choices include transmit power, channel, and antennas. For good performance, reliability and coverage, the physical layer settings should match the RF channel over which the wireless signals are sent. This is evident in rate adaptation schemes that determine the highest rate for transmission, since a good scheme has a large effect on throughput. Other work adapts transmit power to reduce co-channel interference.

In theory, it is simple to select the physical layer configuration because this is directly determined by the specifics of the RF channel. The signal-to-noise ratio (SNR) is the gold standard for performance in narrowband channels. Textbook formulas relate the error rate of different modulations to the SNR. The best rate or required transmit power is then simple to compute. In practice, 802.11 LANs have never used channel measurements as more than a coarse indicator of expected performance. There have simply been too many ways in which the observed measurements and actual performance fail to match the predictions of theory. For example, the most accessible channel measurement is received signal strength indication (RSSI), which serves as a proxy for the true SNR. RSSI measurements are samples that may vary over packet reception, be mis-calibrated, or be corrupted by interference, all of which are known to be issues in practice. Even if RSSI were perfect, it does not reflect the frequency selective fading of 802.11 channels, which are not close to narrowband. Nor does it account for imperfect receivers that may greatly degrade performance. Due to these factors, the minimum RSSI at which a rate starts to work varies by more than 10 dB for real links.

To reconcile these viewpoints, a form of guided search is widely used in practice to select operating points. Packet delivery is simply tested for a rate or transmit power to see how well it works. If the loss rate is too high, a lower rate (or more power) is used, otherwise a higher rate (or less power) is tested. SampleRate is a well-known algorithm of this kind for finding transmit rates. This approach is very effective for slowly varying channels and simple configurations (e.g., a few rates with fixed transmit power and channel) since the best setting will soon be found.

However, search becomes less effective as channels change more quickly and the configuration space becomes more complex. Both of these factors are trends: 802.11 clients are increasingly used when they are truly mobile, both walking and in vehicles; and NICs that are now being deployed with 802.11n depend on multiple antennas, which adds another dimension to and increases the size of the search space. Also, tuning combinations such as rate and power is much more complex. For rate selection, recent work has made headway by measuring symbol-level details of packet reception. In particular, SoftRate uses the output of soft-Viterbi decoding for each symbol to estimate the bit error rate (BER). This allows it to predict the effects on packet delivery of changing the rate. AccuRate uses symbol error vectors for the same purpose. However, these methods are not defined for selecting other useful parameters, such as transmit power, and they do not extend from 802.11a/g to 802.11n, e.g., when selecting antennas or numbers of spatial streams. [4]

III. PROPOSED SYSTEM

Using BER Information on Sender

The BER of the packets contains valuable information about the current wireless carrier, where the carrier can include factors such as the modulation and coding scheme, frequency band, transmission power setting, and routing path. For systems that can use partial packets, this fine-grained and direct BER information enables the sender to better (and adaptively) select a carrier with the best goodput. Here, goodput is defined as the number of recoverable (application-level) bits per second that the system can transmit. The following presents some concrete examples.

Wi-Fi Rate Adaptation: In Wi-Fi networks, a sender has the choice over different data rates. Higher rate means larger number of bits transmitted per second, but also with higher probability of error. Rate adaptation thus aims to select the best data rate, dynamically based on the time-varying wireless channel condition. Previous rate adaptation schemes are often based on coarse-grain information such as packet delivery ratio or indirect information such as SNR. In comparison, the fine-grained and direct BER information provided by EEC enables the sender to better find a rate with the best goodput. The above discussion can also be generalized to using BER information for better selection of wireless channel, transmission power, or directional antenna orientation. **BER-Aware Routing:** In a multihop wireless network, a source can often choose among different routes to send packets to the destination, which can be viewed as one kind of carrier selection. Existing route selection schemes usually consider correct (full) packet delivery only and thus optimize for minimizing the expected number of transmissions (including retransmission of partial packets) needed to deliver the packet. For systems that can use partial packets, one would imagine that the route selection process should instead optimize for maximizing the goodput of the end-to-end route. Obviously, EEC can readily provide the BER information for each wireless link to enable such route selection.

Using BER Information on Receiver

Instead of feeding back the BER information to the sender, the receiver of a partial packet can also directly utilize such information. We focus on scenarios where the receiver itself is an intermediate router in a multihop wireless network. Following are some concrete examples showing how the receiver can use the BER information to make informed decisions when processing a partial packet.

BER-Aware Packet Retransmission: Today, wireless mesh networks have been widely deployed as a cost-effective way to provide Internet access for both urban and rural areas. To enable services such as remote learning and remote healthcare, there are strong demands to support real-time multimedia applications (e.g., video chatting, video conferencing, and VoIP) in these networks. Let us take real-time video streaming as an example. To deal with errors in wireless communication, the source often adds forward error correction on the packets to avoid the extra delay involved in packet retransmission. With forward error correction, the receiver (router) will now simply forward all packets (correct or partial) to the next hop, with the hope that the final destination can recover the partial packets via error correction. However, with the time-varying quality of the wireless links, it is impractical to add sufficient error correction redundancy to ensure that all partial packets can be recovered.

The BER information provided by EEC conveniently enables the receiver (router) to avoid this problem. Namely, the source (knowing the details of the forward error correction applied to the packet) can easily include a threshold in the packet header, indicating the maximum BER that the forward error correction can tolerate. The router can now request retransmission of those packets whose BER exceeds such threshold, instead of naively relaying the packets to the next hop. Of course, an alternative approach would be for the receiver to decode the error correcting code on the partial packet and request retransmission if decoding fails. However, this will require the router to be non oblivious and to know the exact error correcting mechanism employed by the application. Furthermore, as we will show later, the computational overhead of error correcting codes may prevent the router from decoding (in software) at Wi-Fi data rate. **BER-Aware Packet Scheduling:** Consider a wireless image sensor network for emergency response (e.g., forest fire, flood, or earthquake). In such scenarios, the system needs to send back as much information as possible and as fast as possible. For image data, a partially correct packet often still carries useful information, where the information can be a function of the packet's BER. As the data funneling to the base station, the BER information on the packets enables the sensors to

prioritize the forwarding of packets with lower BER. Doing so will maximize the amount of information collected by the base station at any given time point.

BER-Aware Packet Forwarding: In a typical setting of cooperative relay, a dedicated relay node may help one node A to better transmit packets to another node B (within A's radio range). The relay node, within the radio range of both A and B, simply relays the packets that it overhears from A to B. B will eventually combine these (potentially partial) packets. When relaying, the relay node can choose between amplify-and-forward (AAF) and decode-and-forward (DAF). DAF can remove noise before forwarding, but suffers from error propagation if the decoded packet contains many errors. AAF has the opposite property. Researchers thus suggest that ideally the relay should adaptively choose between the two depending on the error level of the packet. The quantitative BER information provided by EEC naturally fits such needs.

Error Estimation Formal Framework

Let denote the total number of data bits in a packet. From the data bits, the EEC encoding process will generate EEC bits for error estimation later. The sender will send these bits in a packet to the receiver. Here, the notion of a packet is logical: It can be an 802.11 packet, or a segment in an 802.11 packet, or multiple 802.11 packets, in which case EEC will estimate the average BER over these multiple 802.11 packets. We model a packet as slots, with each slot holding one bit. A slot may be erroneous and cause the bit in that slot to be flipped during transmission and that flipped bit is called an error. A slot that is not erroneous is called correct. Let denote the fraction of erroneous slots, or equivalently, the BER of the packet. Notice that is a fraction instead of a probability. We only aim to estimate for since, in practice, a packet with BER larger than 1/4 rarely has any value. The errors may be in arbitrary positions in the packet. In particular, the errors may be correlated in an arbitrary and unknown way (e.g., fully clustered or widely spread). The randomization used in the EEC algorithm exactly serves to deal with such (arbitrary) correlation and our algorithm does not assume that the errors are independent.

OTP

In this paper node departure may happen frequently by client leave or failure. By client leave, we mean that the client notifies all its collaborating peers before it actually leaves the session or explicit leave, while by client failure, the client does not notify its collaborating peers when it actually leaves. In this work, the failure of any client will be detected by its peers. The system should exert itself to suppress the propagation of undesirable impact of client leave/failure. Specifically, the receiving peer X must promptly react to this situation by reconfiguring the layer transmission policy because its streaming quality may be degraded during this transient period. In case of losing a supplying peer Pm, for reducing the complexity of reschedule algorithm, the layer allocation on P1 through Pm_1 need not be altered. The reason is they have already been optimized through the OTP algorithm.

GDS:

GDS is proposed in this section. The distinction of node leave and failure is that, the leave node will notify X to reconfigure the transmission policy on its own. Meanwhile, Pm would continue to stream data to X until the reconfigure process has completed. While in case of node failure, the reconfiguration would not be started until the failure is detected. So it would have a churn of playback quality for its downstream nodes. For ease of description, we take use of an example for description. X initially received data from supplying peers P1 through P4. Upon losing a supplying peer P2, X asks other peers to stream. Specifically, P3 is responsible for meeting the layer gap left by P2. However, due to limited uplink bandwidth, providing layers for P3 have to shift down. For the same reason, the supplying layers from P4 are shifted down likewise. Actually, GDS is to group existing peers as a new set of supplying peers for X, and then reinitialize relevant parameters and execute the OTP algorithm again, for maximizing the streaming quality when nodes departure occurred.

IV. RESULTS

The concept of this paper is implemented and different results are shown below, The proposed paper is implemented in Java technology on a Pentium-IV PC with minimum 20 GB hard-disk and 1GB RAM. The propose paper's concepts shows efficient results and has been efficiently tested on different Datasets.

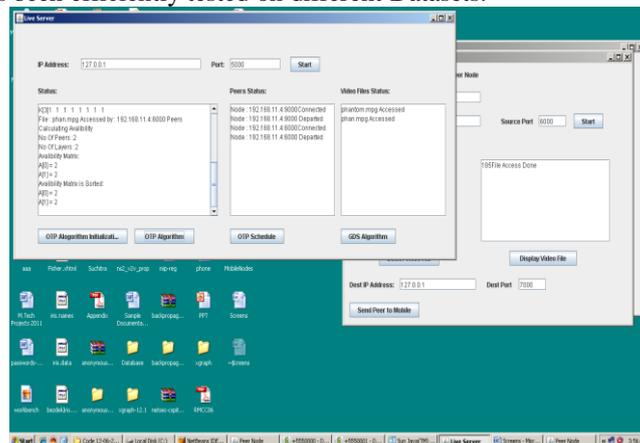


Fig. 1 Proposed system performing video coding.

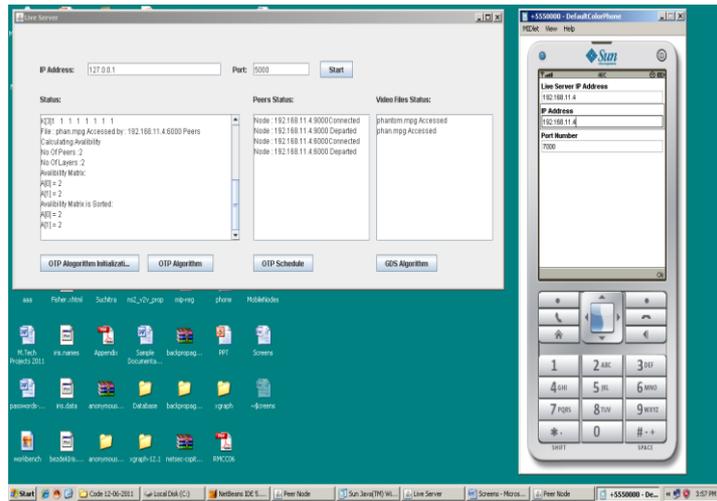


Fig. 1 Proposed system performing communication

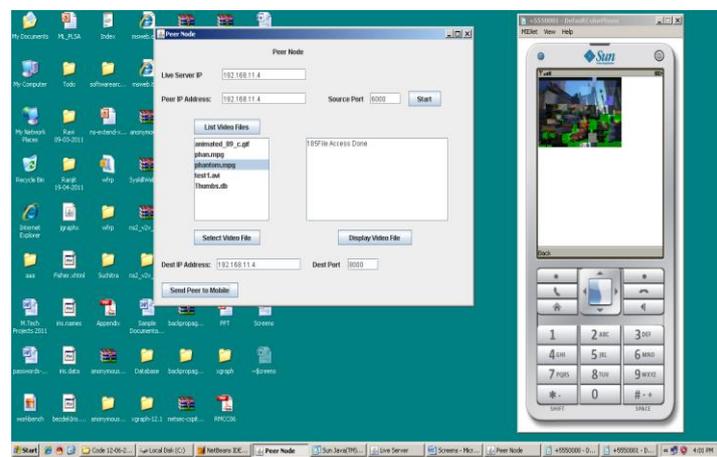


Fig. 3 displaying video performing error correction

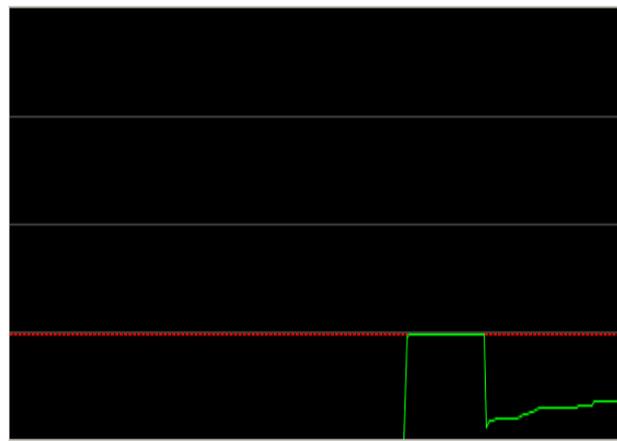


Fig. 4 Time taken by Node to initialize

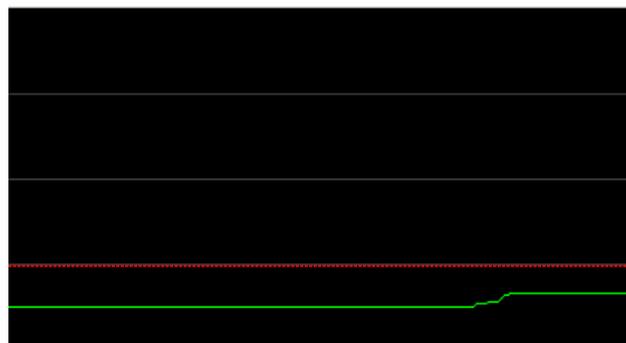


Fig. 5 Time taken by Phase Distribution

V. CONCLUSIONS

This paper is motivated by recent emerging systems that can leverage partial packets in wireless networks. We observe that such systems would significantly benefit from the BER information of the partial packets. This paper thus proposes the novel concept of error estimating coding (EEC). Without correcting the errors, EEC enables the receiver of a partial packet to estimate the packet's BER. Our EEC design provides provable estimation quality, with rather low redundancy and computational overhead. We have exploited and implemented EEC in two wireless network applications, Wi-Fi rate adaptation and real-time video streaming. Our real-world experiments have demonstrated that these applications can significantly benefit from EEC. While we have only focused on applying EEC to wireless networking in this paper, the utility of EEC can be much broader. For example, EEC's functionality can also help data storage recovery from multiple partially correct copies. Generally speaking, EEC may find potential application wherever partially correct data can be utilized.

REFERENCES

- [1] Y. Li and B. Vucetic, "On the performance of a simple adaptive relaying protocol for wireless relay networks," in *Proc. IEEE VTC-Spring*, 2008, pp. 2400–2405.
- [2] X. Liu, A. Sheth, M. Kaminsky, K. Papagiannaki, S. Seshan, and P. Steenkiste, "DIRC: Increasing indoor wireless capacity using directional antennas," in *Proc. ACM SIGCOMM*, 2009, pp. 171–182.
- [3] J. Camp and E. Knightly, "Modulation rate adaptation in urban and vehicular environments: Cross-layer implementation and experimental evaluation," in *Proc. ACM MobiCom*, 2008, pp. 315–326.
- [4] D. Halperin, W. Hu, A. Shethy, and D. Wetherall, "Predictable 802.11 packet delivery from wireless channel measurements," in *Proc. ACM SIGCOMM*, 2010, pp. 159–170.
- [5] B. Chen, Z. Zhou, Y. Zhao, and H. Yu, "Efficient error estimating coding: Feasibility and applications," National University of Singapore, Singapore, Tech. rep., 2010 [Online]. Available: <http://www.comp.nus.edu.sg/~yuhf/eec-tr.pdf>
- [6] G. Judd, X. Wang, and P. Steenkiste, "Efficient channel-aware rate adaptation in dynamic environments," in *Proc. ACM MobiSys*, 2008, pp. 118–131.
- [7] S. Katti, D. Katabi, H. Balakrishnan, and M. Medard, "Symbol-level network coding for wireless mesh networks," in *Proc. ACM SIGCOMM*, 2008, pp. 401–412.
- [8] M. Vutukuru, H. Balakrishnan, and K. Jamieson, "Cross-layer wireless bit rate adaptation," in *Proc. ACM SIGCOMM*, 2009, pp. 3–14.
- [9] H. Yu, "Secure and highly-available aggregation queries in large-scale sensor networks via set sampling," in *Proc. IPSN*, 2009, pp. 1–12.
- [10] K. Tan, J. Zhang, J. Fang, H. Liu, Y. Ye, S. Wang, Y. Zhang, H. Wu, W. Wang, and G. M. Voelker, "Sora: High performance software radio using general purpose multi-core processors," in *Proc. NSDI*, 2009, pp. 75–90.
- [11] M. Vutukuru, K. Jamieson, and H. Balakrishnan, "Harnessing exposed terminals in wireless networks," in *Proc. NSDI*, 2008, pp. 59–72.
- [12] K. Lin, N. Kushman, and D. Katabi, "ZipTx: Harnessing partial packets in 802.11 networks," in *Proc. ACM MobiCom*, 2008, pp. 351–362.