中央研究院
資訊科學研究所
Institute of Information Science, Academia Sinica • Taipei, Taiwan, ROC

TR-IIS-06-010

# Effective File Transfer for Opportunistic Networks

Chen-Hung Yu, Yung-Chih Chen, Ling-Jyh Chen

**August 31, 2006 || Technical Report No. TR-IIS-06-010**

http://www.iis.sinica.edu.tw/LIB/TechReport/tr2006/tr06.html

# Effective File Transfer for Opportunistic Networks*

Chen-Hung Yu[1], Yung-Chih Chen[2], Ling-Jyh Chen[2]

[1]Department of Computer Science and Information Engineering
National Taiwan University
[2]Institute of Information Science
Academia Sinica

August 21, 2006

**Abstract**

As the sheer number of potential opportunistic application continues to surge (i.e., wireless sensor networks, underwater sensor networks, pocket switched networks, vehicular networks, and etc.), proper strategies for dealing with communication in various challenged network environments are of significance and remained desirable. In this study, we investigated two applications in opportunistic networks, namely file transfer and video transfer applications. Based on the H-EC approach, we proposed three message scheduling algorithms to effectively transfer data files in challenged networks. Moreover, targeting video file transfers, we designed LMDC (Layered Multiple Description Coding) based techniques that immensely

* ***This manuscript has been submitted to IEEE Infocom'07.***

improve the perceived video quality for end users. Using simulations as well as realistic network scenarios, we evaluated various proposed schemes in terms of latency and PSNR (Peak Signal-to-Noise Ratio). We showed that our proposed schemes can achieve much better latency performance for file transfers. Furthermore, we show that using LMDC-based techniques, the end user can enable lower quality video "previews" before the video file is completely transferred. The effectiveness and robustness of the proposed schemes render them ideal solutions that can go a long way toward effective file transfer in opportunistic networks.

# 1 Introduction

With wireless networking technologies extending into the fabrics of our working and operating environments, proper handling of intermittent wireless connectivity and network disruptions is of significance. As the foreseeable need of data communication escalates in *challenged* network environments, an increasing amount of attention has been invested to techniques that can address these anticipated requirements. Applications of these techniques are broad. For instance, it would be quite advantageous to interconnect mobile search and rescue nodes in disaster areas (where communication infrastructures are disabled by earthquake, hurricane, wildfire, or flooding), allowing message exchanges in developing areas (remote towns and villages interconnected by wireless networks, but not guaranteed an always-on Internet connection), and permitting scientific monitoring of wilderness (remote monitoring of various wildlife).

Formally, an opportunistic network is a type of challenged networks that satisfies the following conditions: (1) network contacts (i.e., communication opportunities) are intermittent, (2) an end-to-end path between the source and the destination may have never existed, (3) disconnection and reconnection is common, and/or (4) link performance is highly variable or extreme. As a result of various disruptions and/or high delays, traditional MANET and Internet routing techniques can not be directly applied towards networks in this category. With numerous emerging opportunistic networking applications, such as wireless sensor networks (WSN) [6] [35], underwater sensor networks (UWSN) [14], pocket switched networks (PSN) [9] [10] [19], people networks [31] [33], transportation networks [3] [7] [21], and etc., it remains desirable/necessary to develop an effective scheme that can better accommodate the various characteristics/applications of opportunistic networks.

Several data forwarding schemes have been previously proposed for opportunistic networks [7] [17] [22] [25] [32] [34] [37]. Those routing schemes can be grouped into two main categories according to their basic technical strategies, namely *replication based* and *coding based*. In fact, coding based schemes tend to be more robust than the replication based schemes when the network connectivity is extremely poor (this is considered as the *worst delay performance cases*). However, coding based schemes are less efficient when the network is well connected (this is considered as the *very small delay performance cases*), which is simply due to additional information embedded in the code blocks.

Nevertheless, a successful information forwarding scheme in opportunistic networks not only needs to consider delay performance, but it must also consider the nature of its application. Effective schemes dealing with different application requirements remain challenging and desirable. In this study, we target two particular applications in opportunistic networks, namely file transfers and video transfers. Based on H-EC [11], a recently adopted hybrid routing scheme that takes advantages of both replication and erasure coding techniques, we proposed three new *message scheduling* algorithms to enhance the data delivery performance of H-EC in opportunistic network scenarios. Different from previous studies on opportunistic routing, we focus on file transfer scenarios that assume the source has the complete set of messages before initiating data forwarding (rather than generating messages on the fly as employed in [11] [34]). Expanding our proposed message scheduling algorithms to video applications, we also designed new LMDC (Layered Multiple Description Coding) based techniques that can immensely improve end users' perceived video quality and viewing experience. With simulations and realistic network scenarios (based on network traces), we evaluated the various proposed schemes in terms of latency and PSNR (Peak Signal-to-Noise Ratio). The results indicate that our proposed schemes can achieve much better latency performance for file transfers. Specifically, the results indicate that HEC-BI and HEC-SF can provide good performance for networks with good connectivity, and HEC-FI can provide more resilient performance in cases of poor network connectivity. Moreover, we show that with our proposed LMDC-based techniques, the end user can "preview" a video at a lower quality even before the video file is completely transferred, thereby improving the overall viewing experience.

The rest of the paper is organized as follows. In section 2, we summarize related work in opportunistic routing, and recap some commonly used terminologies. Section 3 provides comprehensive overview of the H-EC routing scheme and describes the details of our newly proposed enhancements. In section 4, an LMDC scheme with unequal erasure protection scheme is presented, aiming to allow more efficient video file transfers over opportunistic networks. Section 5 presents a rich set of simulation results from various opportunistic network scenarios; the results are then analyzed and explained in detail. Lastly, section 6 concludes the paper.

# 2 Related Work

Routing in an opportunistic network is challenging and remains unique from conventional network routing methods. An ideal routing scheme in opportunistic networks has to provide reliable data delivery even when the network connectivity is intermittent or when an end-to-end path is temporally nonexistent. Moreover, since ‘contacts’ in an opportunistic network may appear arbitrarily without prior information, neither scheduled optimal routing (e.g., linear programming routing in DTN of scheduled contacts [20]) nor mobile relay approaches (e.g., Message Ferrying [38] [39]) can be applied.

For opportunistic networks, replication is the most popular design choice in existing opportunistic routing schemes. For instance, the *Epidemic Routing* scheme

[32] sends identical copies of a message simultaneously over multiple paths to mitigate the effects of a single path failure, thus increases the possibilities of successful message delivery. However, flooding duplicate data tends to be very costly in terms of traffic overhead and energy consumption.

To address the excess traffic overhead from flooding replicate data, a *Controlled Flooding* scheme has been proposed to reduce the flooding cost while maintaining good reliability in message delivery [17]. In this scheme, the message flooding is controlled by three parameters, namely *willingness probability*, *Time-to-Live*, and *Kill Time*. Additionally, once the receiver successfully receives a message, a *Passive Cure* is generated to "heal" the nodes in the network after they have been "infected" by that message. Therefore, with the ability to resolve excess amount of traffic overhead while provides reliable data delivery, controlled flooding scheme greatly relieves the network overhead .

Node mobility also impacts the effectiveness of opportunistic routing schemes. When the network mobility departs from the well-known random way-point mobility model (e.g., the Pursue Mobility Model [8] and the Reference Point Group Mobility Model [18]), previous studies have shown that the overhead carried by epidemic and/or flooding based routing schemes can be further reduced by taking into account the knowledge of node mobility. For instance, *Probabilistic Routing* scheme [24] calculates the *delivery predictability* from a node to a particular destination node based on the observed contact history, and it forwards a message to its neighboring node if and only if that neighbor node has a higher delivery predictability value. This scheme has also been revised by Leguay *et al.* [22] by taking the *mobility pattern* into account, i.e., a message is forwarded to a neighbor node if and only if the neighbor node has a mobility pattern more similar to the destination. [22] shows that the revised *mobility pattern* based scheme is more effective than previous ones.

Apart from the previous mentioned schemes, another class of opportunistic network routing schemes have been developed via encoding techniques, which transform a message into another format prior to transmission. For instance, an integration of *network coding* and epidemic routing techniques has been proposed to reduce the required number of transmissions in the network [37], and [34] proposes to combine *erasure coding* and the simple replication based routing method to improve the data delivery for the *worst delay performance* cases in opportunistic networks.

Although network coding based routing schemes are promising in reducing the number of transmissions (thus, improve routing efficiency) in a network, it may still fail in providing effective data delivery when the delivery latency is dominated by some extremely large *inter-contact time* (i.e., the time duration between two communication opportunities). In such extreme cases, forwarding schemes based on erasure coding would be more ideal, since the destination is able to reconstruct the message by just receiving a certain number of erasure coded blocks, instead of all transmitted data.

Based on the erasure coding based data forwarding scheme [34], an Estimation based Erasure-Coding routing scheme (EBEC) has been proposed to adapt the delivery of erasure coded blocks using the Average Contact Frequency (ACF) estimate [23]. Moreover, [11] proposes a hybrid scheme to combine the strength of erasure coding and the advantages of *Aggressive Forwarding*, so that it not only remains robust for *worst delay performance cases*, but also performs efficiently for *very small delay performance cases*.

# 3 H-EC: An Erasure Coding based Hybrid Routing Approach

## 3.1 H-EC Overview

In this subsection, we will give a brief overview on erasure code and a forwarding scheme based on erasure code, which is proposed in [34].

Erasure coding is a coding scheme that provides better fault-tolerance by adding redundancy without the overhead of strict replication to the original data [36]. Two most popular erasure coding algorithms are Reed-Solomon coding and Low-Density Parity-Check (LDPC) based coding (e.g., Gallager codes, Tornado codes, and IRA codes) [26] [30]. These algorithms differ in the encoding/decoding efficiency, replication factor, and the minimum number of code blocks to reconstruct a message. The selection of the proper erasure coding algorithm is not within the scope of this paper, and our work is based on the generic erasure coding concept.

In generic erasure coding schemes, suppose a message is of size $M$ bytes, the replication factor of erasure coding is $r$, and the coded message is fragmented into several blocks of identical size $b$ bytes, one can obtain the number of the coded blocks by $N = \frac{M \times r}{b}$. Moreover, this message can be successfully reconstructed as long as $\frac{1}{r}$ of the coded blocks are received, i.e., the minimal number of coded blocks for successfully reconstructing the message is $N/r$.

In [34], an erasure code based forwarding algorithm (EC) is proposed as illustrated in Fig. 1. In this scheme, the erasure coded blocks are equally split among $n$ relays, and relays are only allowed to send messages directly to the destination (i.e., the well-known "two-hop" scenario as used in [9] [16]). Each relay forwards the same amount of code blocks (no duplicates in each relay), and the number of blocks forwarded by each relay can be obtained by[1]

$$\frac{N}{n} = \frac{Mr}{bn} \tag{1}$$

As reported in [34], the EC scheme is capable of providing the best *worst-case delay performance* with a fixed amount of overhead. However, the drawback of EC scheme is that it can not provide good *very small delay performance* while comparing to other popular replication based approaches. The reason for such inefficiency lies in its block allocation method. In EC scheme, the number of transmitting blocks in each contact is a fixed number (i.e., $\frac{Mr}{bn}$, in accordance with Eq. 1) regardless of the length of each contact duration. As a result, EC scheme can only effectively utilize each network contact when the contact duration is slightly longer than the required time for sending the relayed data. If most network contact is much larger than the required time, EC scheme tends to *waste* the residual contact duration and thus results in ineffectiveness as illustrated in Fig. 1.

[1]For simplicity, we assume $N$=$n$, as presented in [11] [34], for all cases.

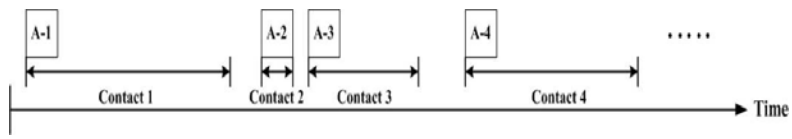



Figure 1: Illustration of the erasure coding based data forwarding algorithm (EC). In this figure, one erasure coded block (A) is equally split among four relays ($n = 4$).

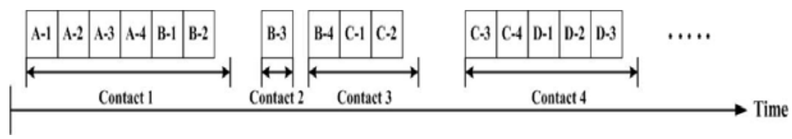



Figure 2: Illustration of the A-EC scheme, i.e., EC with aggressive forwarding. In this figure, four erasure coded blocks (A,B,C,D) are transmitted, and $n = 4$.

Aiming at this problem, [11] has proposed an enhanced scheme called A-EC, i.e., EC with *aggressive forwarding* feature, as shown in Fig. 2. In this scheme, the source sends as many coded blocks as possible during each contact (totally $\frac{Mr}{bn}$ blocks, i.e., $\frac{Mr}{n}$ bytes). Therefore, A-EC scheme has been shown that it is able to better utilize the network contact and thus expected to outperform EC scheme for *very small delay performance* cases.

However, for *worst delay performance* cases, A-EC has been shown to yield poor delivery ratio and/or very large delivery delay when *black-holes*, i.e., the relays are either unreliable (e.g., with very limited battery power and/or buffer size) or hardly moving closer towards the destination, are present in the network [11].

Taking advantages of both EC and A-EC schemes in order to achieve better

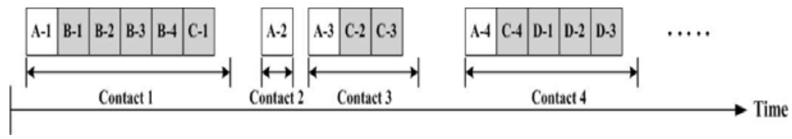



Figure 3: Illustration of H-EC scheme. In this figure, two copies of four erasure coded blocks (A,B,C,D) are transmitted: the first copy of EC blocks (the white blocks) is sent using EC algorithm, and the second copy (the gray ones) is sent using A-EC algorithm in the residual contact duration. Each coded block is equally split into 4 sub-blocks ($n = 4$). This is actually the HEC-SF scheme, we will elaborate more on this in the next subsection.

message delivery performance in both *worst delay performance* and *very small delay performance* cases, a hybrid scheme, called H-EC, is thus proposed. Fig. 3 illustrates the H-EC scheme.

As shown in Fig. 3, in H-EC scheme, two copies of EC blocks (constructed based on the erasure coding and replication techniques described previously) are transmitted by the sender. The first copy of EC blocks is sent similar to how the original EC scheme does (shown as the white blocks in Fig. 3), and the second copy of EC blocks is sent using aggressive forwarding during the residual contact duration after sending the first EC block (shown as the gray blocks in Fig. 3). For general opportunistic network scenarios (i.e., without black-hole nodes), H-EC scheme is expected to better utilize each contact opportunity (i.e., due to aggressive forwarding feature); however, while black-hole nodes are present in the network, H-EC scheme is expected to perform similarly to EC scheme, which provides better forwarding performance in the *worst delay performance* cases.

On the other hand, the performance of H-EC may also highly depend on the employed message scheduling algorithm in the aggressive forwarding phase that is not detailed in [11]. It is one of the interests of this paper to investigate the impact of different message scheduling algorithms on the performance of H-EC routing. We present three message scheduling algorithms in the following subsection.

## 3.2 Message Scheduling in H-EC

In this subsection, we propose three message scheduling algorithms, namely Sequential Forwarding (SF), Full Interleaving (FI), and Block-based Interleaving (BI), for transmitting the second copy of erasure coded blocks (i.e., using aggressive forwarding) in H-EC scheme. For simplicity, we assume the data file to be transferred in an opportunistic network is $L$ messages in size. After applying erasure coding (i.e., adding redundancy), each message becomes $N$ erasure code blocks size. We denote $M_{l,n}$ to index the $n$-th block of the $l$-th message. Moreover, we assume each message can be reconstructed when at least $B$ out of $N$ blocks are successfully received by the destination. We represent the three algorithms as follows.

### 3.2.1 Sequential Forwarding (SF)

In the first algorithm, called Sequential Forwarding (SF), the second copy of erasure coded blocks are sent sequentially in accordance with the order of the messages. The main advantages of this scheme are (a) it is intuitive and easy to implement, and (b) it requires minimal amount of buffer size on the sender side

**Algorithm 1** The algorithm of H-EC with Sequential Forwarding (HEC-SF). The initial values of ($l$, $n$) are (0, 0).

```
Function HEC-SF (l, n)
if l = L and n = N then
    return EndOfMessage
else if l = 0 and n = 0 then
    l ← 1; n ← 1
    return M_{l,n}
else if 1 ≤ l ≤ L and 1 ≤ n < N then
    n ← n + 1
    return M_{l,n}
else if 1 ≤ l < L and n = N then
    l ← l + 1; n ← 1
    return M_{l,n}
else
    return Error
end if
```

(i.e., it does not need to perform erasure coding to all messages in advance). The SF algorithm is detailed in Alg. 1.

### 3.2.2 Full Interleaving (FI)

Different from the Sequential Forwarding (SF) scheme, the Full Interleaving (FI) algorithm proposed to interleave the second copy of the erasure coded blocks for H-EC. More precisely, while doing aggressive forwarding, the FI algorithm transmits the "first" coded block of all the messages at the outset, then the second block of all the messages, and so on and so forth. The advantages of this scheme are (a) it distributes the blocks of each message in a more diverse manner, and is thus expected to be more resilient to *black-hole* scenarios; and (b) since a message can be reconstructed after receiving just a portion of all coded blocks, this scheme

**Algorithm 2** The algorithm of H-EC with Full Interleaving (HEC-FI). The initial values of ($l$, $n$) are (0, 0).

**Function HEC-FI (*l, n*)**
**if** $l = L$ and $n = N$ **then**
  **return** $EndOfMessage$
**else if** $l = 0$ and $n = 0$ **then**
  $l \leftarrow 1$; $n \leftarrow 1$
  **return** $M_{l,n}$
**else if** $1 \leq l < L$ and $1 \leq n \leq N$ **then**
  $l \leftarrow l + 1$
  **return** $M_{l,n}$
**else if** $l = L$ and $1 \leq n < N$ **then**
  $l \leftarrow 1$; $n \leftarrow n + 1$
  **return** $M_{l,n}$
**else**
  **return** $Error$
**end if**

is expected to experience less overall delivery latency than SF algorithm. The FI algorithm is detailed in Alg. 2.

#### 3.2.3 Block-based Interleaving (BI)

The main drawback of FI scheme is the very long response time needed to reconstruct the messages when $L$ and/or $B$ are large (i.e., the time between sending the first block and successfully reconstructing the first message). More specifically, the response time of the FI scheme is definitely greater than the required time to forward the first $L \times (B-1)+1$ blocks[2]. One clever strategy against this problem is to send $B$ blocks during each contact period and interleave the sending process

[2]The response time will become even larger if we also consider data loss, data disorder, and inter-contact time.

**Algorithm 3** The algorithm of H-EC with Block-based Interleaving (HEC-BI). The initial values of $(l, n)$ are (0, 0).

```
Function HEC-BI (l, n)
i ← Int(n/B); j ← n mod B
if l = L and n = N then
   return EndOfMessage
else if l = 0 and n = 0 then
   l ← 1; n ← 1
   return M_{l,n}
else if j! = 0 and n < N then
   n ← n + 1
   return M_{l,n}
else if (j = 0) or (j ≠ 0 and n = N) then
   if l = L then
      l ← 1; n ← (i + 1) × B + 1
   else
      l ← l + 1; n ← i × B + 1
   end if
   return M_{l,n}
else
   return Error
end if
```

among $L$ messages, instead of just sending a single block as described in the FI scheme. The resulting scheme is called the Block-based Interleaving (BI) scheme. Note that, FI scheme is a specialized case of BI scheme with $B$ equal to $N$. The algorithm for the BI scheme is detailed in Alg. 3.

# 4 Layered Multiple Description Coding (LMDC) Video with Unequal Erasure Protection

In this section, we propose the use of Layered Multiple Description Coding (LMDC) with unequal erasure protection [13] for video file transfer in opportunistic networks.

LMDC has been proposed to combine Multiple Description Coding (MDC) [15] and Layered Coding [27] for emerging multicast and peer-to-peer audio/video streaming applications. More specifically, multiple descriptions are striped across multiple packets (or paths) via MDC, and transmitted to a collection of clients, thereby ameliorating the loss of packets due to network congestion or the failure of unreliable hosts. Applications of MDC are IP level multicast [12] and application-level multicast [28] [29]. Moreover, by Layered Coding, multimedia data can be encoded into different quality levels, and the clients would play the most adequate video/audio quality level depending on their capabilities (such as screen resolution, link bandwidth, and etc.).

Combining MDC and Layered Coding, LMDC scheme strips the layered video across multiple packets with multiple descriptions, and the clients are allowed to play the layered video as long as a required portion of descriptions are successfully received. Of course, the more descriptions a client receives, the better the reconstructed video quality. In practice, the LMDC scheme is usually implemented in conjunction with Unequal Erasure Protection [13]. We illustrate the conjunct scheme in Fig. 4.

As illustrated in Fig. 4, the quality of a layered video frame increases as the size of the collected video bit stream increases. More specifically, suppose one of the layered video frames is $S_k$ bits in size, one can split the video frame into $k$ equal-sized pieces and reconstruct the video frame to $Q_i$ quality level by using any $i$ out of the $k$ pieces (i.e., the required bit stream size for reconstructing $Q_i$ level frame is $S_i = i \times S_k/k$).

Each layered video frame is then spilt among $N$ packets ($N \geq k$) with unequal erasure protection on each frame piece. For instance, the $i$-th piece of the layered video frame is erasure coded with replication factor $r$ equal to $i$ and is split among $N$ packets (i.e., the $i$-th piece of the video frame can be reconstructed by any $i$ out of the $N$ packets). The size of the $i$-th coded frame piece, $b_i$, can therefore be obtained by Eq. 2, and the size of the resulting $N$ packets, $b_{packet}$, can be obtained by Eq. 3. Moreover, comparing to Layered Coding scheme, the traffic overhead of LMDC scheme, $b_{overhead}$, can be obtained by Eq. 4.

$$b_i = (S_i - S_{i-1}) \times \frac{N-(i-1)}{N} = \frac{S_k}{k}(1 - \frac{i-1}{N}) \tag{2}$$

$$b_{packet} = \sum_{i=1}^{k} \frac{S_k}{k}(1 - \frac{i-1}{N}) = S_k(1 - \frac{k-1}{2N}) \tag{3}$$

$$b_{overhead} = Nb_{packet} - S_k = (\frac{2N-k-1}{2})S_k \tag{4}$$

Note that, since $N \geq k$ and $k$ is a positive integer (i.e., $k \geq 1$), one can conclude that (a) $b_{overhead} = 0$ when $N = k = 1$ (i.e., no LMDC); and (b)

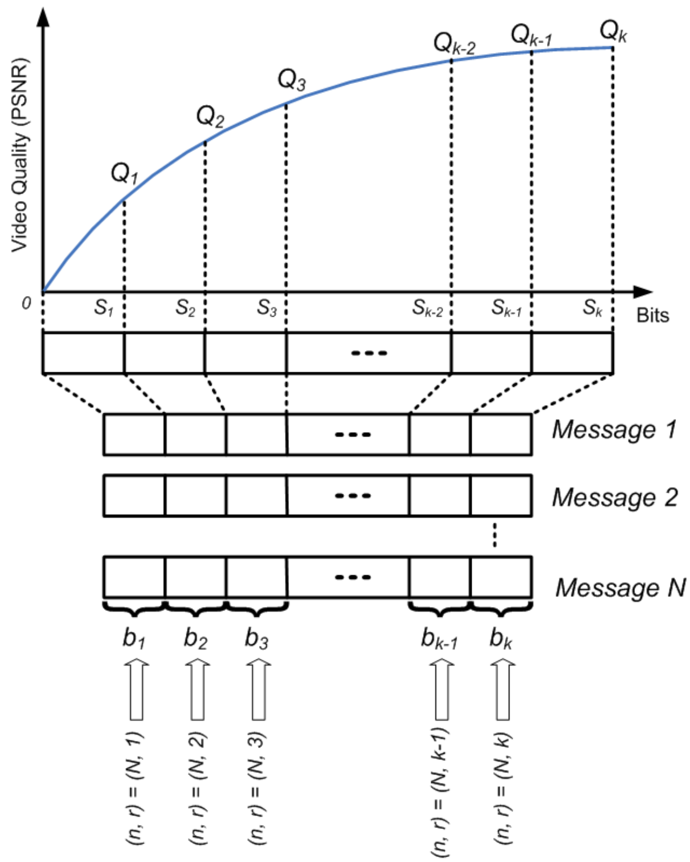
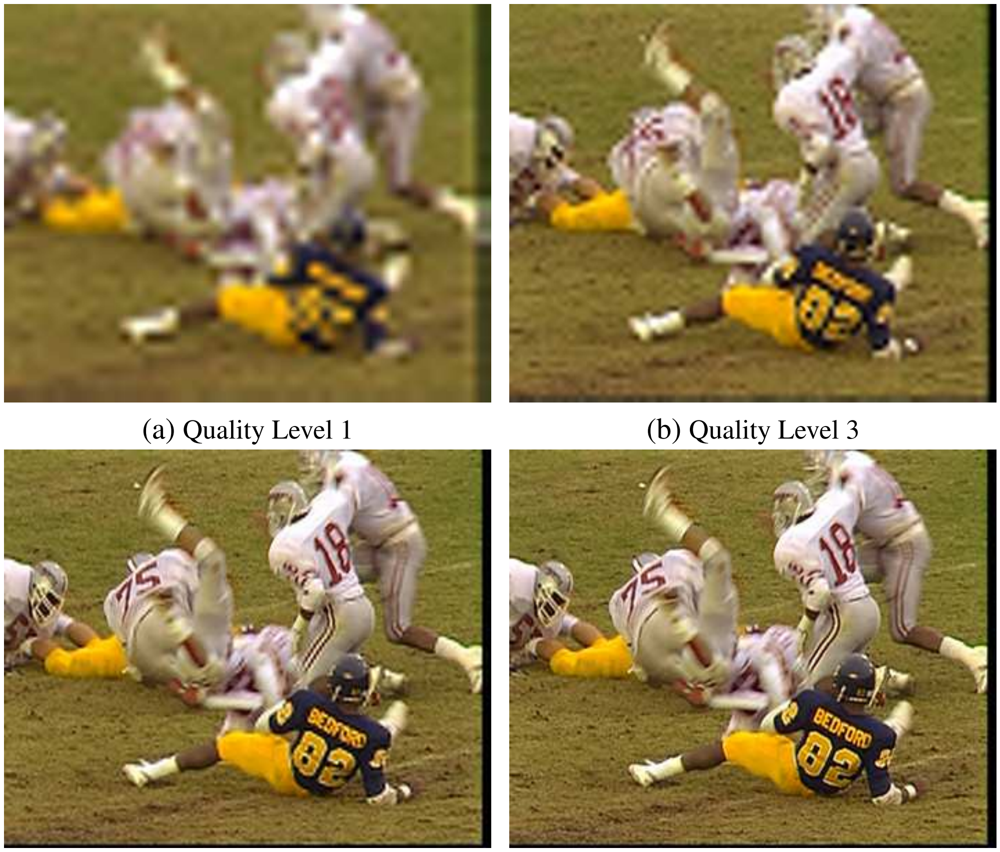



Figure 4: Illustration of Layered MDC scheme with unequal erasure protection: each video frame is encoded into $k$ quality levels using layered coding, and the $i$-th quality level video frame is erasure protected with replication factor $i$ and equally split among $N$ relays ($N \geq k$).




(a) Quality Level 1 (b) Quality Level 3

(c) Quality Level 6 (d) Quality Level 10

Figure 5: An example of LMDC video frame ($k$=10) at various quality levels.

$b_{overhead} > 0$ otherwise. Fig. 5 shows an example of an LMDC video frame at four different quality levels, i.e., $Q_1$, $Q_3$, $Q_6$, and $Q_{10}$, ($k$=10 and $N$=10). From this example illustration, it is clear that the perceived video frame quality significantly improves as the quality level increases.

# 5 Evaluation

In this section, we evaluate the delay performance of file transfer in opportunistic networks. We implemented EC, HEC-SF, HEC-FI, and HEC-BI schemes and performed simulations in DTNSIM [2], a java based DTN simulator. We applied Layered Coding on a 2000-frame video clip using JPEG2000 [4] codec, and we also added unequal erasure protection to each video layer in order to have the video become LMDC encoded. In the simulation, the number of video quality levels, $k$, and the resulting packets for each frame, $N$, are both set to 10. We also assumed that data transmission is error-free at a fixed rate of 1Mbps. The simulation results presented in this section are all obtained by taking the average performance of 200 simulation runs, and in each simulation run the source and the destination pair is randomly selected from all participating nodes.

Three network scenarios are examined in the evaluation. One of them is generated according to the power-law distribution setting both inter-contact time and contact duration of the network with power-law distributed values of coefficient 0.6 (as reported in [9] [19]), and the scenario consists of 34 participating nodes. The other two scenarios are based on realistic campus wireless network traces (namely Dartmouth [1] and UCSD [5] traces), which are publicly released for research references. Table 1 outlines the basic properties of the three network scenarios examined[3].

More specifically, the UCSD trace is a client-based trace that records the vis-

[3]In the network trace provided by Dartmouth, there were a total of 13,888 devices in the network, but only 5,148 of them have contact experience with other devices.

Table 1: Properties of various opportunistic network scenarios.

| Trace Name | Power-Law | UCSD | Dartmouth |
|---|---|---|---|
| Device | N/A | PDA | WiFi Adapter |
| Network Type | N/A | WiFi | WiFi |
| Duration (days) | 16 | 77 | 1,177 |
| Devices participating | 34 | 273 | 5,148 |
| Number of contacts | 25,959 | 195,364 | 172,308,320 |
| Avg # Contacts/pair/day | 2.89205 | 0.06834 | 0.01105 |

ibility of WiFi based access points (APs) with each participating portable device (e.g., PDAs and laptops) on UCSD campus. The network trace is about two and half months long, and there are 273 devices participated. Similar to [9] [10] [19], we make the assumption that a communication opportunity (i.e., network contact) is encountered between two participating devices (in ad hoc mode) if and only if both of them are associated to the same AP during some time period.

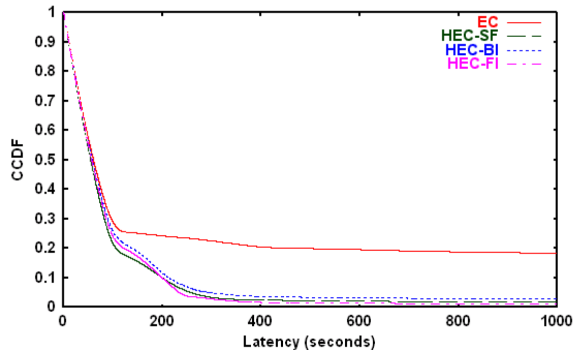
Similarly, the Dartmouth trace is an interface-based trace that records the APs that have been associated with a particular wireless interface during a three year (1177 days) period. However, it should be noted that, wireless interfaces can be used by different devices at different times, and each device may use multiple wireless interfaces. For simplicity, we assume each network interface represents a single mobile user in the network. Moreover, like in the UCSD scenario, a network contact is encountered when two mobile users are associated to the same access point. Note that, although the Dartmouth trace is a lengthier trace with a greater number of participating mobile nodes, the network connectivity is actually very poor in the Dartmouth scenario, since network contacts (for each source-destination pair) occur much more infrequently in the network (nearly one sixth of the UCSD scenario and 0.4% of the Power-Law scenario).

## 5.1 Evaluation: Data File Transfer

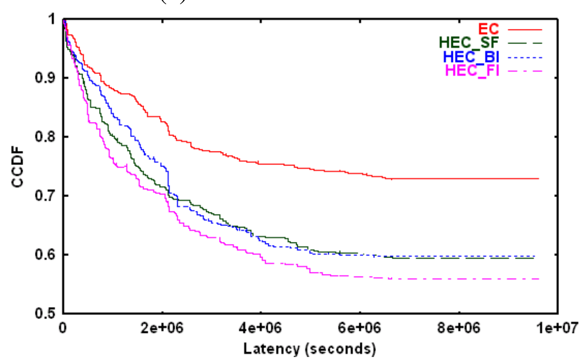
In the first set of simulation, we evaluate the delay performance of our proposed message scheduling algorithms (i.e., HEC-SF, HEC-FI, and HEC-BI) for data file transfer in opportunistic networks. Three network scenarios are examined via simulations, and a huge data file (i.e., 100MBytes in size) is selected for the file transfer. For all employed schemes, the erasure coding parameters ($r$, $n$) are set to (2, 16), which is consistent with the settings employed in [11] [34]. The simulation is performed 200 times for each scheme, and Fig. 6 depicts the average data latency distribution results in Complementary CDF (CCDF) curves.

From Fig. 6, the results clearly show that H-EC based schemes (i.e., SF, BI, and FI) outperform EC scheme in almost all test cases, which further affirms the results from our previous studies [11]. Particularly, the three variants of H-EC perform nearly identically in the Power-Law scenario, and they are able to successfully transfer more than 96% of the data file in the simulation, compare to the EC scheme that is only capable of delivering around 80%.

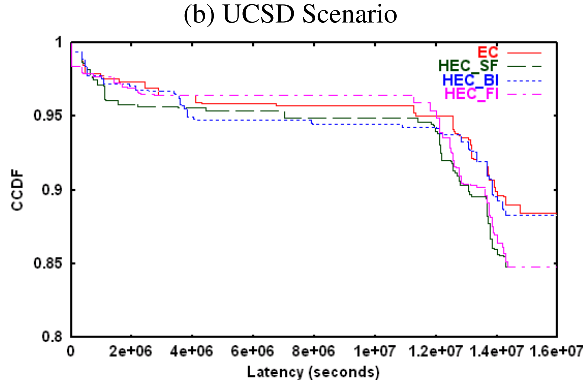
However, it is also observed that the *completion ratio* degrades as the network connectivity (i.e., average number of network contacts per node pair and per day) decreases. For instance, while H-EC based schemes can reach about 96% completion ratio in Power-Law scenario, they can achieve only about 45% in UCSD scenario and 15% in Dartmouth scenario. It turns out that it is quite difficult to




(a) Power-Law Scenario




(b) UCSD Scenario




(c) Dartmouth Scenario

Figure 6: Distribution (CCDF) of average latency performance of EC, HEC-SF, HEC-FI, and HEC-BI schemes ($N = 16$ and $r = 2$).

have a complete data file transfer in an opportunistic network (or it might require a very large file transferring time that is much longer than the employed simulation scenario). Therefore, from the end users' point of view, transferring data file in an opportunistic network may not be feasible unless the receiver has the ability to read out *partial* information from an uncomplete file. This finding inspires us in our investigation into video file transfer with LMDC technique. We present the evaluation of LMDC based video file transfer in the next subsection.

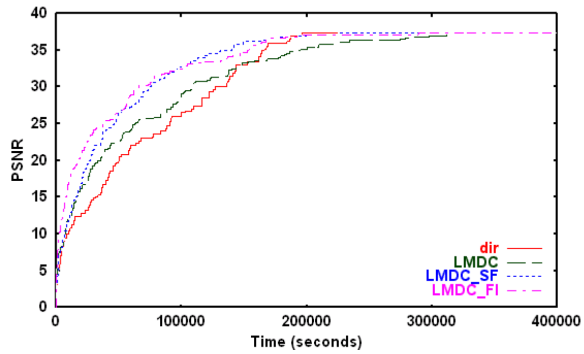
Additionally, one can find that HEC-SF and HEC-BI schemes perform similarly in all scenarios, and consistently perform better in the beginning of the three examined scenarios. This is in contrast to the HEC-FI scheme, which only performs better after a certain latency value. These results seem to contradict our initial intuition that HEC-BI should have the best performance at all times. The reason is because HEC-SF and HEC-BI schemes can both achieve better *contact efficiency* since they allow the receiver to reconstruct the original message as long as a certain number of coded blocks are received; whereas the interleaving nature of the HEC-FI scheme requires the receiver to wait at least a few contacts (depending on the erasure coding parameter, $r$) before it can collect sufficient coded blocks to reconstruct the message. As a result, as depicted in Fig. 6, HEC-SF and HEC-BI usually perform better (more aggressively) on the left portion of the figure (i.e., the small latency cases that represent good network connectivity), and HEC-FI usually performs better (more resiliently) on the right portion in the figure (i.e., the worst latency cases that represent poor network connectivity).

## 5.2 Evaluation: Video File Transfer

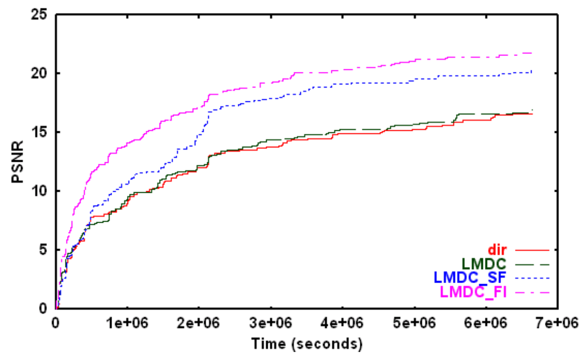
In the second set of evaluation, we investigate the performance of video file transfer (in terms of Peak Signal-to-Noise Ratio, PSNR) in opportunistic networks with and without LMDC based coding schemes. Note that, the LMDC (with unequal erasure protection) scheme is in fact identical to the EC scheme, except that LMDC employs multiple redundancy levels (i.e., $r$ parameter) for each individual video quality layer, instead of just using one redundancy level for the whole message. Moreover, we apply the concepts of HEC-SF/FI algorithms to LMDC scheme by sending the second copy of LMDC blocks via the remaining contact duration just as in the HEC-SF/FI scheme, and called resulting schemes LMDC-SF and LMDC-FI respectively. It should also be mentioned that there is no LMDC-BI scheme in our evaluation since $N = k$, i.e., the minimal number of required blocks for reconstruct the original quality video is exactly the same as the number blocks transferred over the networks[4].

We only evaluated the performance of video file transfer in Power-Law and UCSD scenarios, since the network connectivity of Dartmouth scenario is very sporadic and the data delivery in that scenario is very poor as illustrated in the previous subsection. For the Power-Law and UCSD scenarios, we took the average performance results of 200 runs, and in each run, we randomly select one node as the video source and one node as the video destination. The video file is again 2000-frames in length, and all simulation parameters (e.g., transmission

[4]When $N > k$, one can derive LMDC-BI scheme by simply applying HEC-BI algorithm with block number set equal to $k$.



(a) Power-Law Scenario



(b) UCSD Scenario

Figure 7: Average video quality of LMDC video transfer using H-EC SI, FI, and BI schemes. ($N$=16, and the replication factor of the employed erasure coding, $r$, is 2)

rate and erasure coding parameters) are kept the same as the previous subsection. Fig. 7 shows the average PSNR performance of the 2000-frame video file using different coding and forwarding schemes.

In Fig. 7, it is as expected that the H-EC based schemes (i.e., LMDC-SF and LMDC-FI) would outperform the LMDC and the *dir* (employs *direct contact* algorithm [34] to transfer video file directly without LMDC coding) schemes.

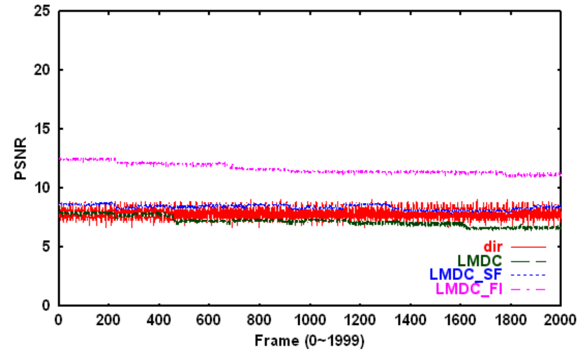
More precisely, LMDC-FI scheme would performs comparatively better than the LMDC-SF scheme in the simulation. This is because the FI based strategy combines both aggressive forwarding and interleaving techniques that not only make aggressive use of precious network contacts, but also does their best to alleviate the negative effects caused by possible black-holes[5].

Additionally, LMDC scheme actually performs similarly (with slight performance gain) to *dir* scheme. This result does contradict with our intuitive understanding of LMDC scheme's added resilience to error-prone and/or poorly-connected networks [13]. It turns out, the reason for the similarity in performance is due to the amount of overhead carried by LMDC (i.e., layered coding and erasure protection). The overhead is too costly to actually provide performance gain in extremely challenged networks scenarios.

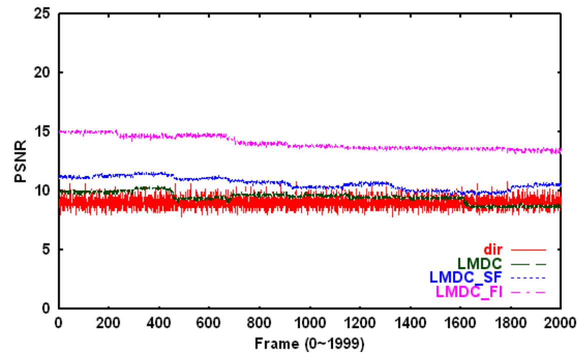
On the other hand, instead of looking at the overall average PSNR of the video, it is also important to look at the frame-by-frame PSNR performance of the video file transfer, since the variance of the frame-by-frame PSNR can also greatly impact the perceived video quality for the end users. Fig. 8 shows the frame-by-frame PSNR performance of three time points (500,000, 1,000,000, and 5,000,000 seconds) in the UCSD scenario.

From Fig. 8, it is clear that the frame-by-frame PSNR quality consistently improves as each encoding/forwarding scheme is given more time (also presented in Fig. 7). Moreover, we also noticed that the LMDC-FI scheme consistently outper-

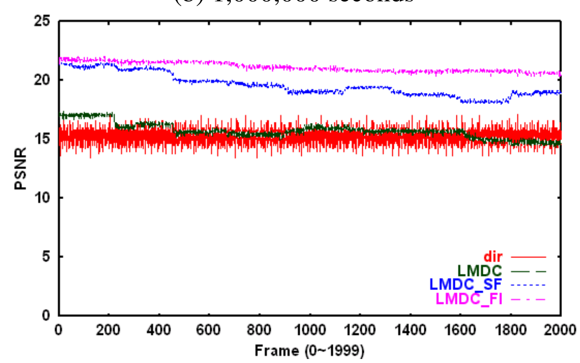
[5]A node is called a *black-hole* in the network if it is either unreliable (e.g., with very limited battery power and/or buffer size) or hardly moving closer towards the destination [11].




(a) 500,000 seconds




(b) 1,000,000 seconds




(c) 5,000,000 seconds

Figure 8: Comparison of average video quality (i.e., PSNR for each video frame) after 500000, 1000000, and 5000000 seconds in the UCSD scenario.

forms the other schemes in all three selected time points; whereas the LMDC-SF scheme performs similarly to the LMDC and the *dir* schemes at the beginning, but it noticeably outperforms those two schemes after 1,000,000 seconds. The results confirm that the *aggressive forwarding* phase is able to significantly enhance the performance of data forwarding in opportunistic networks, and the use of interleaving technique is useful for spreading LMDC blocks over the network, which alleviates the influence of potential black-holes.

Additionally, it is also observed from the figure that, for all LMDC based schemes, the average PSNR value slightly degrades as the frame number increases. This is because these schemes are all basically sending video frames (or say coded blocks) by frame orders, regardless whether it is in the first regular EC sending phase or in the second aggressive forwarding phase. This problem can be easily solved by sending video frames by a uniformly random order; however, the tradeoff in this instance is the resulting computation needed and memory space overhead required. We will defer the detail discussion on this issue to our future work.

It should also be noted that in Fig. 8, the PSNR value of the *dir* scheme oscillates heavily in terms of frequency and amplitude, whereas the curves of LMDC based schemes are much smoother (especially for the LMDC-FI scheme). This is because, in the *dir* scheme, each video frame is either successfully received or completely lost; there is no intermediate quality video that can be played by the end user. As a result, the *dir* scheme tends to yields a large variation in its per-frame PSNR performance. As indicated in [40], such drastic PSNR variation is detrimental to the end users' perceived video quality. Therefore, based on our observations above, the LMDC based schemes can indeed yields higher frame-by-frame PSNR performance for video transfer in opportunistic networks, and they are also capable of providing better perceived quality video to the end users.

# 6 Conclusion

An effective data networking scheme is essential for opportunistic networks, as communication opportunities in such challenged networks are precious and opportunistic in nature. In this paper, we proposed three message scheduling algorithms that extend the data delivery capabilities of H-EC. Expanding these three algorithms to video application, we also designed improvements to Layered Multiple Description Coding (LMDC) that can immensely improve the perceived video quality to end users. With simulations as well as realistic network traces, we evaluated the various proposed schemes. We first show that our proposed schemes can achieve much better latency performance for file transfers. Specifically, the results indicate that HEC-BI and HEC-SF can provide good performances for networks with good connectivity, and HEC-FI can provide more resilient performances in cases of poor network connectivity. Moreover, we show that with our proposed LMDC-based techniques, the end user can "preview" a video at a lower quality before the video file is completely transferred, thereby improving the overall viewing experience.

# 7 Acknowledgements

We gratefully acknowledge the use of wireless data from the CRAWDAD archive at Dartmouth College (sponsored by the National Science Foundation under award number 0454062) and the UCSD Wireless Topology Discovery Trace project (sponsored by DARPA Contract N66001-01-1-8933). This work is sponsored by the National Science Council under grant number NSC-95-2221-E-001-025.

# References

[1] Crawdad project. http://crawdad.cs.dartmouth.edu/.

[2] Delay tolerant network simulator. http://www.cs.washington.edu/homes/sushjain/pubs2/dtnsim.tar.gz.

[3] The disruption tolerant networking project at umass. http://prisms.cs.umass.edu/diesel/.

[4] Jpeg 2000. http://www.jpeg.org/jpeg2000/.

[5] Ucsd wireless topology discovery project. http://sysnet.ucsd.edu/wtd/.

[6] The zebranet wildlife tracker. http://www.princeton.edu/ mrm/zebranet.html.

[7] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine. Maxprop: Routing for vehicle-based disruption-tolerant networking. In *IEEE Infocom*, 2006.

[8] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless Communication and Mobile Computing Journal*, 2(5):483–502, 2002.

[9] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Pocket switched networks: Real-world mobility and its consequences for opportunistic forwarding. Technical Report UCAM-CL-TR-617, University of Cambridge, Computer Laboratory, February 2005.

[10] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott. Impact of human mobility on the design of opportunistic forwarding algorithms. In *IEEE Infocom*, 2006.

[11] L.-J. Chen, C.-H. Yu, T. Sun, Y.-C. Chen, and H. hua Chu. A hybrid routing approach for opportunistic networks. In *ACM SIGCOMM Workshop on Challenged Networks*, 2006.

[12] P. A. Chou and K. Ramchandran. Clustering source/channel rate allocations for receiver-driven multicast under a limited number of streams. In *IEEE ICME*, 2000.

[13] P. A. Chou, H. J. Wang, and V. N. Padmanabhan. Layered multiple description coding. In *IEEE Packet Video Workshop*, 2003.

[14] J.-H. Cui, J. Kong, M. Gerla, and S. Zhou. Challenges: Building scalable mobile underwater wireless sensor networks for aquatic applications. *IEEE Network, Special Issue on Wireless Sensor Networking*, May 2006.

[15] V. K. Goyal. Multiple description coding: Compression meets the network. *IEEE Signal Processing Magazine*, pages 74–93, September 2001.

[16] M. Grossglauser and D. Tse. Mobility increases the capacity of ad-hoc wireless networks. In *IEEE Infocom*, 2001.

[17] K. A. Harras, K. C. Almeroth, and E. M. Belding-Royer. Delay tolerant mobile networks (dtmns): Controlled flooding in sparse mobile networks. In *IFIP Networking*, 2005.

[18] X. Hong, M. Gerla, R. Bagrodia, and G. Pei. A group mobility model for ad hoc wireless networks. In *ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, 1999.

[19] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot. Pocket switched networks and human mobility in conference environments. In *ACM SIGCOMM Workshop on DTN*, 2005.

[20] S. Jain, K. Fall, and R. Patra. Routing in a delay tolerant network. In *ACM SIGCOMM*, 2004.

[21] J. LeBrun, C.-N. Chuah, and D. Ghosal. Knowledge based opportunistic forwarding in vehicular wireless ad hoc networks. In *IEEE VTC Spring*, 2005.

[22] J. Leguay, T. Friedman, and V. Conan. Dtn routing in a mobility pattern space. In *ACM SIGCOMM Workshop on Delay Tolerant Networks*, 2005.

[23] Y. Liao, K. Tan, Z. Zhang, and L. Gao. Estimation based erasure-coding routing in delay tolerant networks. In *International Wireless Communications and Mobile Computing Conference*, 2006.

[24] A. Lindgren and A. Doria. Probabilistic routing protocol for intermittently connected networks. Technical report, draft-lindgren-dtnrg-prophet-01.txt, IETF Internet draft, July 2005.

[25] A. Lindgren, A. Doria, and O. Schelen. Probabilistic routing in intermittently connected networks. *ACM SIGMOBILE Mobile Computing and Communications Review*, 7(3):19–20, July 2003.

[26] M. G. Luby, M. Mitzenmacher, M. A. Shokrollahi, and D. A. Spielman. Efficient erasure correcting codes. *IEEE Transactions on Information Theory*, 47(2):569–584, February 2001.

[27] S. R. McCanne. *Scalable Compression and Transmission of Internet Multicast Video*. PhD thesis, University of California, Berkeley, 1996.

[28] V. N. Padmanabhan, H. J. Wang, and P. A. Chou. Resilient peer-to-peer streaming. In *IEEE ICNP*, 2003.

[29] V. N. Padmanabhan, H. J. Wang, P. A. Chou, and K. Sripanidkulchai. Distributing streaming media content using cooperative networking. In *IEEE NOSSDAV*, 2002.

[30] J. S. Plank and M. G. Thomason. A practical analysis of low-density parity-check erasure codes for wide-area storage applications. In *The International Conference on Dependable Systems and Networks*, 2004.

[31] D. Snowdon, N. Glance, and J.-L. Meunier. Pollen: using people as a communication medium. *Elsevier Computer Networks*, 35(4):429–442, February 2001.

[32] A. Vahdat and D. Becker. Epidemic routing for partially-connected ad hoc networks. Technical Report CS-2000-06, Duke University, 2000.

[33] R. Y. Wang, S. Sobti, N. Garg, E. Ziskind, J. Lai, and A. Krishnamurthy. Turning the postal system into a generic digital communication mechanism. In *ACM SIGCOMM*, 2004.

[34] Y. Wang, S. Jain, M. Martonosi, and K. Fall. Erasure coding based routing for opportunistic networks. In *ACM SIGCOMM Workshop on Delay Tolerant Networks*, 2005.

[35] Y. Wang and H. Wu. Dft-msn: The delay fault tolerant mobile sensor network for pervasive information gathering. In *IEEE Infocom*, 2006.

[36] H. Weatherspoon and J. D. Kubiatowicz. Erasure coding vs. replication: A quantitative comparison. In *The 1st International Workshop on Peer-to-Peer Systems*, March 2002.

[37] J. Widmer and J.-Y. L. Boudec. Network coding for efficient communication in extreme networks. In *ACM SIGCOMM Workshop on Delay Tolerant Networks*, 2005.

[38] W. Zhao, M. Ammar, and E. Zegura. A message ferrying approach for data delivery in sparse mobile ad hoc networks. In *ACM MobiHoc*, 2004.

[39] W. Zhao, M. Ammar, and E. Zegura. Controlling the mobility of multiple data transport ferries in a delay-tolerant network. In *IEEE Infocom*, 2005.

[40] M. Zink, O. Künzel, J. Schmitt, and R. Steinmetz. Subjective impression of variations in layer encoded videos. In *IWQoS*, 2003.