TRANSMISSION-DISTORTION TRADEOFFS IN NETWORK CHANNEL CODING

Shirish Karande, Hayder Radha[+]
Department of Electrical and Computer Engineering, Michigan State University
East Lansing, MI 48824, USA
Email: {karandes, radha}@egr.msu.edu

ABSTRACT

Network Channel Coding (NCC) is a framework under which intermediate router/nodes employ encoding/decoding operations to facilitate an efficient multicast delivery of video. The network usage and video distortion is a function of the channel coding rates assigned to nodes in the network. In this paper, we investigate the tradeoff between the total bandwidth usage and a global distortion measure. We propose a dynamic programming based framework to identify the optimal *transmission-distortion* operating points. The proposed optimal allocation of coding rates is compared with an earlier version of NCC, Network Embedded FEC (NEF), which optimally places codecs of a fixed channel coding rate within the network. The proposed NCC scheme is shown to achieve significantly improved transmission distortion tradeoffs.

`Index Terms—` Channel coding, TV,

1. INTRODUCTION

In a multi-hop communication, the losses over each hop can accumulate to often render the redundancy in an end-to-end Forward Error Correction (FEC) scheme insufficient to facilitate any data recovery. This is especially true for large networks, due to the large number of packet losses at deep nodes of the underlying distribution tree. Meanwhile, emerging network paradigms, such as overlay, peer-to-peer, and ad-hoc wireless networks, open the door for new approaches of delivering low-latency video to a large number of receivers while maintaining high level of reliability and throughput. Significant improvement in performance can be obtained if channel coding functionalities are incorporated optimally at intermediate nodes. Network Channel Coding (NCC) presents a framework to distribute channel coding functions over network nodes, so that data recovery within the network can be utilized to avoid prohibitive accumulation of losses. A particular case of NCC is our prior work [1]-[2], Network Embedded FEC (NEF), which has been shown to exhibit significant improvements in the performance of video streaming applications over multicast peer-to-peer (p2p) trees. In NEF, Reed-Solomon (RS) channel codecs with a pre-determined coding rate are embedded optimally in the distribution tree to maximize throughput. The NEF framework despite its proven utility is fairly restrictive when compared with the proposed NCC framework.

We motivate the proposed investigation, by observing that capacity deductions are typically based on infinite code-lengths and usually for short finite block-lengths the impact of error-exponents cannot be ignored. Consequently, often for both end-to-end FEC as well as NCC, the reliability can be improved to a virtual 100% only by utilizing a prohibitively high redundancy overhead. In practice, this overhead has to be maintained below a maximum level due to bandwidth constraints emerging from a variety of factors. In case of the wired networks discussed in this paper, the factors could be congestion, monetary costs etc. Consequently a crucial problem to be investigated is to identify the optimum network channel coding "choices" that can achieve the best throughput (measured over all receivers) while minimizing the overall network overhead.

In this work, we propose a unifying framework under which a NCC solution will be characterized by a set of permissible forwarding policies, Π. Thus, NEF will be represented by a specific choice of Π. The performance of a NCC solution is determined by rule $\Omega : V \to \Pi$, which assigns a forwarding/coding policy to the nodes in the network. For a given network topology, channel conditions and video source, the coding assignment Ω determines, the average number of total packet transmission per message block $T(\Omega)$ and, the average distortion $D(\Omega)$ perceived by the clients. In the NCC problem we are primarily interested in identifying the optimal *Transmission-Distortion* (TD) curve, which consists of the operating points that represent the best possible tradeoff between $T(\Omega)$ and $D(\Omega)$.

This remainder of this paper is organized as follows: In section 2 we introduce some essential notation and abstractions. In section 3 we state the NCC-NEF and NCC-RAN problems in terms of the proposed abstraction and also state a generic iterative greedy algorithm that can be used to solve the considered NCC problems. In section 4, we consider some sample network topologies, channel conditions and video sources to illustrate the performance benefits of NCC-RAN over NCC-NEF.

2. NOTATION AND NETWORK MODEL

In our generic framework we assume that the network topology is represented by a directed (rooted) tree $T(V,E)$.

[+] This work is supported by NSF Awards CCF-0515253, CNS-0430436, and MEDC Grant GR-296.

$V=\{v_1,\cdots v_{|V|}\}$ is the set of the vertices in the graph such that the indices are assigned in a topologically sorted manner, thus the vertex v_1 represents the source. $E=\{e_1,\cdots,e_{|E|}\}$ is the set of directed edges (all directed away from the root v_1). $U=\{u_1,\cdots u_{|U|}\}\subset V$ represent the set of sink nodes. We associate with each vertex v_z:

- ***children*** **(C_z):** A set of children C_z.
- ***parent*** **(p_z):** Parent p_z s.t. $\forall$ z $\neq 1$, we have $(p_z, v_z)\in E$

Additionally, with the network we associate the constants:

- ***message_block_length*** **(*k*):** Within the framework of NCC we assume that video content is transmitted in batches of k message packets
- ***maximum_code_length*** **(*N*):** With each node we can associate a block-length parameter such that the total number of message packets transmitted by the node for each message block is always less than $n(v)$. We assume that the maximum value of the block-length parameter $n(v)$ can be bounded by a suitably chosen constant N.
- ***quality_vector*** **(Y):** The quality of received data is a function of the number of received packets per message block. Thus $Y[y_i]_{N\times 1}$ is a column vector such that y_i represents the expected quality of video when *i* packets are received per message block.
- ***transmission_penalty_vector*** **(H):** The cost of network usage is a function of the number of packet transmissions. Thus $H[h_i]_{N\times 1}$ is a column vector such that h_i represents the cost of transmitting *i* packets per message block. Note that depending on our definition of network usage we can modify H to represent the complexity associated with decoding/encoding, monetary costs etc.
- ***link_matrices*****:** With each edge in the set E we associate a *link matrix* L_e to characterize the link-loss behavior of the edge. For all $L_e[l_{ij}(e)]_{N\times N}$, $l_{ij}(e)$ represents the probability of *j* packets being successfully received when *i* packets were transmitted. Our link-loss model assumes that the losses over all the links are independent. However the packet losses over a single link may exhibit temporal correlation.
- ***forwarding_policies*** **(Π):** Π is a set of $N\times N$ matrices that represent the permissible encoding functionalities that can be assigned to the vertices. We denote an element of Π as $F[f_{ij}]_{N\times N}$, where f_{ij} represents the probability of a node forwarding j packets when it receives i packets. We currently only consider cases where the set Π satisfies the following properties

i. We assume $|\Pi|$ is finite. Thus the elements of Π can be indexed and represented as $\Pi=\{F_1,\cdots F_{|\Pi|}\}$.

ii. We assume a condition of strict monotonicity such that $\forall i,j \;\; 1\le i,j,\le|\Pi|,\; i<j \Leftrightarrow F_i \prec F_j$. Here $F_i \prec F_j$ implies that when F_i is replaced by F_j the total network usage and total video throughput both increase. When $F_i \prec F_j$ we refer to F_j as a policy greater than F_i (note this is not necessarily a better choice).

Our definition of the set Π is generic enough to describe a wide variety of channel coding (as well as retransmission) schemes. Never the less, due to brevity, we shall consider the following two policy sets determined on the basis of FEC schemes employing Reed-Solomon (RS) like codes [3].

Network Embedded FEC (NEF)**:**

$$\Pi^{NEF}_{\Delta} := \{I_{N\times N}\}\cup\left\{F\left[f_{ij}\right]\left|\begin{aligned}&f_{ij}=u[k-1-i]\cdot\delta[i-j]\\&+u[i-k]\delta[k+\Delta-j]\end{aligned}\right.\right\} \quad \text{where } \Delta\ge 0$$

Also note that $u[\cdot]$ and $\delta[\cdot]$ are the discrete unit step and impulse functions. Thus, the fixed rate policy Π^{NEF}_{Δ} consists of:

- A forwarding policy represented by a matrix $I_{N\times N}$ implies that a node is a pure forwarding node.
- A policy where if a node receives more than $k-1$ packets then it creates additional Δ parity packets to transmit a total of $k+\Delta$ packets.

Our work on NEF [1]-[2] had employed such fixed rate policies. The term NEF is appropriate for a set Π^{NEF}_{Δ} because it consists of only one forwarding policy that employs channel coding/decoding.

Rate-Adjustable Network Channel Coding (RAN)**:**

$$\Pi^{RAN}_{\Delta} := \left\{F_\kappa\left[f_{ij}\right]\left|\begin{aligned}&f_{ij}=u[k-1-i]\cdot\delta[i-j]\\&+u[i-k]\delta[\kappa\cdot\Delta-j]\end{aligned}\right.\right\} \text{ where } 0\le\ \kappa\cdot\Delta<N$$

The set Π^{RAN}_{Δ} consists of policies that transmit additional $\kappa\cdot\Delta$ parity packets whenever atleast *k* packets are received. Thus changing a forwarding policy is equivalent to changing κ and thus the rate of the underlying RS code. Hence we can refer to the set Π^{RAN}_{Δ} as Rate Adjustable NCC (RAN). Note that, as per the above definition of Π^{RAN}_{Δ}, if an intermediate node receives less than *k* packets then it does not add any additional parity and merely forwards all the received packets. Also note that $\left|\Pi^{RAN}_{\Delta}\right| = \left\lfloor\dfrac{N-k-1}{\Delta}\right\rfloor$

forwarding_assignment_rule **(Ω):**

An assignment rule $\Omega: V\to\Pi$, assigns a forwarding policy to each vertex. Note that the assignment rule Ω can be represented by an $|V|$-dimensional integer vector $\Omega=\{a_{1\Omega},\cdots,a_{i\Omega},\cdots,a_{|V|\Omega}\}$ which implies that a forwarding matrix $F_{a_{i\Omega}}$ has been assigned to vertex *i*. Note that for a given policy set there exist $|V|^{|\Pi|}$ operating points. Thus an arbitrary policy assignment is unlikely to provide desired performance.

3. OPTIMIZATION FRAMEWORK FOR NCC

Our optimization objective is to identify the convex-hull of the loci defined by operating points $(T(\Omega), D(\Omega))$. We adopt the following dynamic programming approach which allows us to characterize almost the entire TD curve upto a $T_{\max}$ (or $D_{\min}$) in a single run. In practice we observe that the TD curve provided by the following algorithm is sufficiently close to the actual convex-hull.

Dynamic Programming Algorithm **(***Algorithm 1***):**

The dynamic programming algorithm proceeds by improving the assignment rule Ω in a greedy manner. Let's denote by $\Gamma^{+}(\Omega)$ the set of assignment rules that are obtained by changing just one of the forwarding functionalities to a "greater" functionality, i.e. $\Gamma^{+}(\Omega) = \left\{ \Omega_i \middle| j \neq i \Leftrightarrow \mathrm{a}_{j\Omega_i} = \mathrm{a}_{j\Omega}, j = i \Leftrightarrow \mathrm{a}_{j\Omega_i} = \mathrm{a}_{j\Omega} + 1 \right\}$. Thus our dynamic programming algorithm can be described as follows:

Step 1: Initialize: $\Omega := [1,\ldots,1]_{1\times|V|}$ and $TD = \{(T(\Omega), D(\Omega))\}$

Step 2a: Update: $\Omega := \arg\min_{\Omega_i \in \Gamma(\Omega)} \left(\frac{D(\Omega_i) - D(\Omega)}{T(\Omega_i) - T(\Omega)} \right)$ (greedy step)

Step 2b: $TD := TD \cup (T(\Omega), D(\Omega))$

Step 3: If $T(\Omega) \leq T_{\max}$ (OR $D(\Omega) \geq D_{\min}$) go to step 2

Note that step 2 in the above algorithm choose the *steepest-decent* direction. In the above algorithm a decent direction is guaranteed due to the monotonic nature of the set $|\Pi|$.

The above algorithm can be easily made to resemble the greedy algorithm proposed in [1] by modifying the step 2(a) as

(*Algorithm* 2**):**

Step 2a: Update: $\Omega := \arg\min_{\Omega_i \in \Gamma(\Omega)} \left(D(\Omega_i) - D(\Omega) \right)$ (greedy step)

Thus in this work we shall compare the TD curve provided by Algorithm 1, when operated on Π_{Δ}^{RAN}, with the TD curve provided by Algorithm 2, when operated on Π_{Δ}^{NEF}.

Evaluation of Transmission/Distortion**:**

Till this point we have provided a high-level description of the proposed optimization framework. However, we haven't as yet, explicitly provided a mechanism to evaluate the transmission and distortion for a given policy assignment. Thus to provide such mechanism we associate the following vector with each vertex in the graph:

- ***reliability_vector*** **(**$R_z\left[r_i^z\right]_{N\times 1}$**):** A variable row vector R_z such that r_i^z determines the probability with which node v_z receives *i* packets.

Thus the average quality provided to a receiver v_z is given by:

$$Q_z(\Omega) = R_z Y$$

We associate a constant $Q_{\max}$ with a given video source and *N*, such that $Q_{\max}$ represents the best possible quality that can be provided a client. On the basis of the above measure of quality, the global distortion measure is given by:

$$D(\Omega) = Q_{\max} - Q(\Omega) = Q_{\max} - \frac{1}{|U|} \sum_{z\, s.t\ v_z \in U} R_z Y \qquad (1)$$

In this work, we assume a wired multicast network model and thus average cost per message block incurred for node v_z is given by

$$T(\Omega) = |C_z| R_z F_{a_z} H$$

Consequently the total network usage is given by

$$T(\Omega) = \left(\sum_z |C_z| R_z F_{a_z} H \right) \qquad (2)$$

Evaluation of Reliability Vectors**:**

From the equations (1) and (2) it should be obvious that if we know the reliability vector for each vertex then for a given assignment Ω, $T(\Omega)$ and $Q(\Omega)$ can be evaluated in time $\mathrm{O}(|V|)$. The reliability matrices themselves can be evaluated in time $\mathrm{O}(|V|)$, using the following algorithm

(*Algorithm* 3):

FOR z = 1: $|V|$,

IF $z = 1$ then R_z is $s.t\ r_i^z = \delta[k - i]$

ELSE $R_z := R_{p_z} F_{a_{p_z\Omega}} L_{(p_z, z)}$

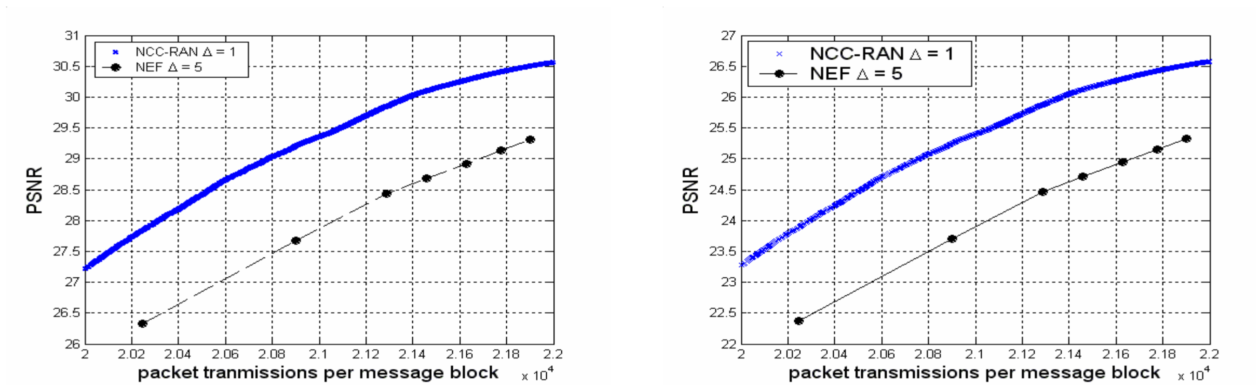
where $L_{(p_z, z)}$ is the link matrix associated with an edge coming from a node's parent. Also note that the above algorithm requires that the vertices are indexed in a topological order.

The above presented algorithm if applied directly leads to a time-complexity of $O\left(|V|^2\right)$. However this time complexity can be easily brought down to $O(|V|)$ by storing some partial calculations. However due to brevity description of such algorithmic optimizations is outside the scope of this paper.

4. RESULTS AND ANALYSIS

The dynamic programming framework described in the previous sections is based on abstract objects and thus can be used to optimize a variety of NCC scenarios.

- The tree structure can be chosen to represent a variety of networks. We are currently exploring topologies emerging from Content Distribution Networks (CDN) like Akamai [4], various p2p topologies discussed in [5], etc. Never the less due to brevity, in this paper, for the sake of illustration we consider an arbitrary randomly evolving tree of 1000 nodes [2], where incoming peers join any existing peer in a uniformly random manner. However we constrain the maximum degree of each node to 3.
- A link matrix can be utilized to represent a variety of channels. Thus the work presented in [1] is now meagerly a special case of the above described framework. In principle



(a) Carphone (b) Paris

Figure 1 Transmission-Quality Curves for NEF and NCC-RAN

we can derive the link matrix on the basis of actual network traces [6] thus significantly improving the practical validity of our conclusions. However for sake of illustration, here, we restrict ourselves to link matrices that are determined by a Bernoulli random variable with the loss probability 0.03.

- Appropriate choice of quality matrices allows us to incorporate the properties of specific video/multimedia source. In the sample results we present here, the quality matrices have been derived on the basis of actual experiments with H.264 TML 9.0 [7] compressed video streams. Quality matrices for test sequences “carphone” and “paris” were made by measuring the quality as a function of the number of packet drops per message block. We empirically evaluated these matrices by randomly dropping packets in a video sequence. The matrices are formed by averaging the performance over 30 repetitions of such a experiment. The compression parameters we employed were: a GOP of size 15 frames, with structure IPP..; a QP of 28 and resolution “CIF”.
- For all the simulations in this section we utilize a simple definition of H given by $h_i = i$, i.e. the network usage cost is measured purely in terms of the total number of packet transmissions.

In addition to the above restriction, for the purpose of illustration , we assume k=20, $\Delta = 5$ for NEF and a step size $\Delta = 1$ for RAN.

Figure 1 exhibits the Transmission-Quality Tradeoff Curve for NEF and NCC-RAN. From Figure 1 it can be seen that an NCC-RAN scheme can provide the same quality of service as NEF while utilizing a 1000 fewer packet transmissions. In addition, it can be observed that NEF provides us with only 7 operating points when $T_{\max}$ is limited to 22000 packet transmissions per block. As against that NCC-RAN provides us with over 800 operating points within the range of 20000-22000 packet transmissions. There exist several more operating points where fewer than 20000 packet transmissions are used. Thus clearly our generic framework allows us to analyze and develop schemes like NCC-RAN which can provide significantly improved NCC schemes. The performance of NCC-RAN scheme as compared to NEF is observed to improve substantially in presence of network heterogeneity (i.e. scenarios where different links in the network are experiencing varying channel conditions). However due to brevity we are unable to include any additional results.

5. CONCLUSIONS

In this paper we proposed a generic abstract framework, Rate-adjustable NCC (RAN), within which we optimize the channel coding rates allocated to intermediate nodes within the network. We provided a mechanism to systematically identify the (near) optimal Operational Transmission-Quality curve for the considered NCC schemes. It was observed that the proposed RAN scheme provides an average gain in excess of 1dB over NEF. More importantly RAN provides a richer set of operating points which, in future, could be suitably exploited for time-varying channel conditions or demands.

REFERENCES

[1] M. Wu, S. Karande and H. Radha, "Network Embedded FEC for Optimum Throughput of Multicast Packet Video," EURASIP Signal Processing: Image Communication - Special Issue on Video Networking, vol. 20, no. 8, pp. 728-742, September 2005.

[2] S. Karande, H. Radha, "Influence of Graph Properties of Peer-to-Peer Topologies on Video Streaming with Network Channel Coding," ICME, 2006.

[3] L. Rizzo, "Effective Erasure Codes for Reliable Computer Communication Protocols", Computer Communication Review, 1997.

[4] http://www.akamai.com/en/html/services/media_delivery.html

[5] D. Loguinov, J. Casas, and X. Wang, "Graph-Theoretic Analysis of Structured Peer-to-Peer Systems: Routing Distances and Fault Resilience," IEEE/ACM Transactions on Networking, vol. 13, no. 5, October 2005.

[6] M Yajnik, S. B. Moon, J. F. Kurose, D. F. Towsley, “Measurement and Modeling of the Temporal Dependence in Packet Loss”, INFOCOM 1999: 345-352S.

[7] ISO/IEC JTC 1/SC29/WG11 and ITU-T SG16 Q.6, Draft ITU-T Recommendation and Final Draft International Standard of Joint Video Specification (ITU-T Rec.H.264 | ISO/IEC 14496-10 AVC), Doc.JVT-G050, March 2003.-