Degree Distribution Optimization in Raptor Network Coding

Nikolaos Thomos and Pascal Frossard
Signal Processing Laboratory (LTS4)
Swiss Federal Institute of Technology, Lausanne (EPFL), Lausanne, Switzerland
{nikolaos.thomos,pascal.frossard}@epfl.ch

***Abstract*— We consider a multi-source delivery system, where Raptor coding at sources and linear network coding in overlay nodes work in concert for efficient data delivery in networks with diversity. Such a combination permits to increase throughput and loss resiliency in multicast scenarios with possibly multiple sources. The network coding operations however change the degree distribution in the set of packets that reach the receivers, so that the low complexity decoding benefits of Raptor codes are unfortunately diminished. We propose in this paper to change the degree distribution at encoder, in such a way that the degree distribution after network coding operations recovers a form that leads to low complexity decoding. We first analyze how the degree distribution of the encoded symbols is altered by network coding operations and losses in a regular network. Then we formulate a geometric optimization problem in order to compute the best degree distribution for encoding at sources, such that the decoding complexity is low and close to Raptor decoders' performance. Simulations show that it is possible to maintain the low complexity decoding performance of Raptor codes even after linear network coding operations, as long as the coding at sources is adapted to the network characteristics.**

***Index Terms*— Network coding, Raptor codes, degree distribution, geometric programming.**

I. Introduction

Mesh networks are very interesting for data distribution as they typically offer several transmission paths between servers and clients. These paths might share common links and nodes, which motivates the design of appropriate distribution and coding strategies that can properly exploit the network diversity. Network coding (NC) [1] takes advantage of network nodes computation capabilities to increase network throughput and enhance transmission robustness in networks with diversity. The network nodes typically perform coding operations with the received data before forwarding them to next hop nodes. When combined efficiently with coding at sources, network coding leads to distributed data delivery solutions without the need for reconciliation among nodes nor differential treatment of data packets with different importance. This permits to benefit from the network diversity for both increased throughput and improved resilience to losses.

In this paper, we consider the transmission framework represented in Fig. 1. We build on the algorithm that has been proposed in [2] in the specific context of video streaming in overlay mesh networks. We consider that the source symbols are first processed by Raptor encoders [3] with a degree distribution Ω. Raptor codes are chosen due to their rateless performance as well as their linear encoding and decoding times. The encoded symbols are then sent on a regular mesh network, where intermediate nodes implement linear network coding. In case of packet loss, the network nodes generate substitute packets by combining pairs of packets, which are selected in order to maximize the symbol diversity [2]. Due to losses and network coding in successive stages, the degree distribution of the Raptor encoders at sources is altered; it might finally differ significantly from the degree distribution of typical Raptor codes that are characterized by low-complexity decoding. In particular, the generator matrix becomes denser due to combinations of Raptor symbols by network coding, which leads to increased decoding complexity.

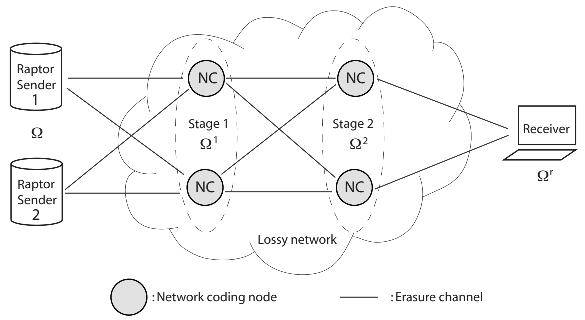

Fig. 1. Illustration of a transmission framework with network coding of Raptor symbols.

The computational complexity of the decoding operations at receivers are clearly driven by the degree distribution of the received symbols. We thus propose to re-design the degree distribution of the encoders at sources such that the actual degree distribution of the receiver permits low-complexity decoding. We propose here a generic method for computing an appropriate source degree distribution for any target degree distribution function after linear network coding in regular topologies. The optimization problem is formulated as a geometric programming (GP) [4] problem. GP is typically characterized by objective and constraint functions of special forms. Furthermore, it is well adapted for solving large-scale optimization problems. We assume that the servers are aware of the network statistics (network losses and topology), and we design efficient source degree distributions so that the

This work has been supported by the Swiss National Science Foundation, under grant PZ00P2-121906.

linear decoding time property is preserved. We show through simulations that the resulting codes only suffer from a minimal performance loss compared to optimal degree distributions at decoder and still provide effective resiliency to losses and network dynamics. A related problem has been studied in [5] where the design of LT codes degree distribution for simple relay topologies has been investigated. The robust soliton distribution (RSD) [6] is decomposed into to two component distributions prior to RSD deconvolution. It preserves the spikes of RSD that guarantee the success of the decoding process. Although the algorithm in [5] ensures that clients receive symbols whose degree distribution is close to RSD, it imposes rather complicated encoding rules. Furthermore, the extension of this method to complex network topologies is non trivial.

II. Analysis of low-complexity Raptor network codes

We analyze here in details the Raptor network coding scheme proposed in [2] for regular networks that have the same number of nodes per coding stage as shown in Fig. 1 and each node is connected with all nodes in the previous coding stage. By Raptor coding at the senders, the data symbols are first pre-coded and then fed into a LT encoder. The transmitted symbols are produced by sampling a degree distribution function

$$\Omega(x) = \sum_{i=1}^{L} \Omega_i \cdot x^i,$$

which determines the number of original data symbols that has been combined for generating the Raptor encoded symbols. Ω_i denotes the probability of generating a symbol with degree i. Since Ω_i's are degree distribution function coefficients the following constraints should hold:

$$\begin{array}{l} \sum\limits_{i=1}^{L} \Omega_i = 1 \\ \Omega_i \geq 0, \forall i \in [1, \ldots, L] \end{array} \tag{1}$$

As the packets travel through the network, they are combined to compensate for packet losses due to erasures and network variations. The network coded symbols follow a new denser degree distribution function that is different from the original function used at the servers. The degree distribution at the k_{th} coding stage is denoted as

$$\Omega^k(x) = \sum_{i=1}^{L} \Omega_i^k \cdot x^i$$

where Ω_i^k, $i = 1, \ldots L$, are the coefficients of the degree distribution function at nodes in the k_{th} coding stage. The degree distribution evolves as the symbols traverse the different encoding stages, and eventually lead to the degree distribution Ω^r observed at the client (decoder).

We now analyze in more details the evolution of the degree distribution in regular networks where all network links have equal capacity and identical loss rate p. First, single hop transmission is examined and then the findings are extended for larger networks. At the first network coding stage, the re-encoded packets correspond to a degree distribution function $\Omega^1(x)$ given by

$$\begin{aligned} \Omega^1(x) &= \sum_{i=1}^{L} \Omega_i^1 \cdot x^i = (1-p) \cdot \sum_{i=1}^{L} \Omega_i \cdot x^i + \\ &+ p \cdot \sum_{i=1}^{L} \Psi_i^1 \cdot x^i = \sum_{i=1}^{L} \left\{ (1-p) \cdot \Omega_i + p \cdot \Psi_i^1 \right\} \cdot x^i \end{aligned} \tag{2}$$

where $\Psi^1(x)$ is the convolution $(\Omega * \Omega)(x)$ since the symbols that are combined follow the degree distribution $\Omega(x)$. It can be written as

$$\Psi_i^1 = \frac{\sum\limits_{\mu+\nu=i} \Omega_\mu \cdot \Omega_\nu}{\sum\limits_{i=1}^{L} \sum\limits_{\mu+\nu=i} \Omega_\mu \cdot \Omega_\nu}, \ i = 0, \ldots, L \tag{3}$$

or equivalently

$$\Psi_i^1 = \frac{\sum\limits_{j=1}^{i-1} \Omega_j \cdot \Omega_{i-j}}{\sum\limits_{l=1}^{L} \sum\limits_{j=1}^{l-1} \Omega_j \cdot \Omega_{l-j}}, \ i = 0, \ldots, L \tag{4}$$

The Ψ_i^1 values are normalized by $A = \sum\limits_{l=1}^{L} \sum\limits_{j=1}^{l-1} \Omega_j \cdot \Omega_{l-j}$ since some symbols combinations are not eligible as the degree of the re-encoded symbols can not exceed L, which is equal to the number of source symbols.

Obviously, the distribution $\Omega^1(x)$ can be seen as the weighted sum of $\Omega(x)$ and $\Psi^1(x)$ with weight parameter p (*i.e.*, the packet loss rate). For the degree distributions of typical Raptor codes such as 3GPP [7] and RSD, the denominator of Eq. (4) is approximately written as

$$\begin{aligned} \sum_{l=1}^{L} \sum_{j=1}^{l-1} \Omega_j \cdot \Omega_{l-j} &= \sum_{l=1}^{L/2} \Omega_i^2 + 2 \cdot \sum_{l=1, l \neq L-l}^{L} \Omega_l \cdot \Omega_{L-l} = \\ &= \left(\sum_{l=1}^{L} \Omega_i \right)^2 - \sum_{l=L/2+1}^{L} \Omega_i^2 = 1 - 0 = 1 \end{aligned}$$

The term $\sum\limits_{l=L/2+1}^{L} \Omega_i^2$ tends to zero as $\sum\limits_{l=1}^{L/2} \Omega_i \approx 1$ for the 3GPP and RSD distributions that are concentrated into small degree values. Thus, the coding coefficients Ω_i^1 of Eq. (2) can be written as

$$\Omega_i^1 = (1-p) \cdot \Omega_i + p \cdot \sum_{j=1}^{i} \Omega_j \cdot \Omega_{i-j}, \text{ for } i = 0, \ldots, L \tag{5}$$

Unfortunately 3GPP Raptor and RSD degree distributions contain spikes that do not permit direct deconvolution. Therefore, the resulting equation system cannot be solved as some of the constraints of Eq. (1) are violated. In order to explain this limitation, we rewrite Eq. (5) as

$$\Omega_i = \frac{1}{1-p} \cdot \{\Omega_i^1 - p \cdot \sum_{j=1}^{i-1} \Omega_j \cdot \Omega_{i-j}\}, \text{for} \quad i = 1, \ldots, L \quad (6)$$

In the case of the RSD, it holds that $\Omega_2^1 > \Omega_1^1$ and $\Omega_{i+1}^1 < \Omega_i^1$, $i = 1, \ldots, M-1, M+1, \ldots L$ where $M = K/S_1$. The K and S_1 are respectively the number of source symbols and a parameter that controls the size of the ripple in every step of the decoding procedure. The ripple contains the symbols that have been recovered but not processed yet. We have the following lemma, whose proof is given in the Appendix.

Lemma 1: If $\Omega^1(x)$ has a spike at Ω_M^1 then it should be:

$$c > \max_{j\in[2,M-2]} \left\{2 \cdot \frac{\Omega_{M-1}}{\Omega_{M-2}}, \frac{\Omega_{M-j}}{\Omega_{M-j-1}}\right\}$$

$$\textit{and} \quad c^{'} > \max_{j\in[2,M-2]} \left\{\frac{1-p}{2 \cdot p \cdot \Omega_1}, \frac{\Omega_{M-j}}{\Omega_{M-j+1}}\right\}$$

for $\Omega_M^1 > 0$. $c, c^{'} \gg 1$ determine the magnitude of the spike and it holds $\Omega_M^1 = c \cdot \Omega_{M-1}^1$ and $\Omega_M^1 = c^{'} \cdot \Omega_{M+1}^1$.

Therefore, it is not possible to find a function $\Omega(x)$ that preserves the spike at the M_{th} position and satisfies the conditions of Eq. (1), since Ω_i's become negative when c and $c^{'}$ take large positive value. Similar conclusions can be drawn for other degree distributions with spikes. Therefore, we can only design suboptimal degree distributions $\Omega(x)$ that lead to final degree distribution $\Omega^r(x)$ that are close but not equal to the target degree distribution functions such as RSD or 3GPP.

III. Degree distribution optimization

We propose now to optimize the degree distribution at the source by an iterative algorithm. If there is only one coding stage, the optimization problem can be formulated as

$$\min_{\{\Omega_i\}} \sum_{i=1}^{L} \left\{ \frac{\Omega_i^1}{(1-p) \cdot \Omega_i + p \cdot \sum_{j=1}^{i-1} \Omega_j \cdot \Omega_{i-j}} \right\} \quad (7)$$

where $\Omega^1(x)$ is the final distribution. The optimization conditions are given in Eq. (1).

We cast this optimization problem into a Geometric Programming (GP) problem. The general form of a GP problem is

$$\min f_0(W)$$

subject to

$$\begin{aligned} f_i(W) &\leq 1 \quad \text{for} \quad i = 1, \ldots, m \\ g_i(W) &= 1 \quad \text{for} \quad i = 1, \ldots, n \end{aligned}$$

where g_i, $i = 1, \ldots, n$ and f_i, $i = 1, \ldots, m$ are respectively monomials and generalized posynomials [8]. $W = (W_1, \ldots W_n,)$ is a vector with n real variables.

To transform Eq. (7) into a valid GP form, we use the geometric inequality

$$\sum_{i=1}^{L} \Omega_i \geq L \cdot \prod_{i=1}^{L} \Omega_i^{\frac{1}{L}}$$

Thus, the objective function is written as

$$\min_{\{\Omega_i\}} \sum_{i=1}^{L} \left\{ \frac{\Omega_i^1}{i \cdot (1-p)^{1/i} \cdot \Omega_i^{1/i} \cdot p^{\frac{i-1}{i}} \cdot \prod_{j=1}^{i-1} (\Omega_j \cdot \Omega_{i-j})^{1/i}} \right\}$$

while the optimization constraints are

$$\begin{aligned} 0 \leq \Omega_i \leq 1 \\ L \cdot \prod_{i=1}^{L} \Omega_i^{1/L} = 1 \end{aligned}$$

Since we are aware of the expected loss rate, we add two new constraints

$$\begin{aligned} \Omega_i^1 &\leq (1+p) \cdot \Omega_i \\ \Omega_i^1 &\geq (1-p) \cdot \Omega_i \end{aligned}$$

These constraints limit the search space and allow fast convergence. The resulting problem can then be solved with help of software packages such as [9], [10].

When we have two coding stages, we consider the optimization problem as a cascade of two dependent GP problems. Since we know the desired degree distribution, we first solve

$$\min_{\{\Omega_i\}} \sum_{i=1}^{L} \left\{ \frac{\Omega_i^2}{i \cdot (1-p)^{1/i} \cdot (\Omega_i^1)^{1/i} \cdot p^{\frac{i-1}{i}} \cdot \prod_{j=1}^{i-1} \left(\Omega_j^1 \cdot \Omega_{i-j}^1\right)^{1/i}} \right\}$$

then after determining the Ω_i^1 coefficients, we successively solve

$$\min_{\{\Omega_i\}} \sum_{i=1}^{L} \left\{ \frac{\Omega_i^1}{i \cdot (1-p)^{1/i} \cdot \Omega_i^{1/i} \cdot p^{\frac{i-1}{i}} \cdot \prod_{j=1}^{i-1} (\Omega_j \cdot \Omega_{i-j})^{1/i}} \right\}$$

to find the original degree distribution. The same process is repeated iteratively for larger networks.

IV. Simulation results

We analyze the performance of the above design algorithm in a Raptor network coding system. We study the performance of a system where we target a RSD-like degree distribution function at decoder, with the design procedure described above. This distribution is used along with a 3GPP pre-coder at the encoder. We compare this system with a variant that combines the same pre-coder as 3GPP codes and LT codes with RSD distribution (denoted as RSD). The performance is compared in terms of decoding probability and decoding complexity. For the sake of completeness, we also study the performance of classical 3GPP degree distribution (denoted as 3GPP), which are heuristic distributions commonly used in wireless broadcasting systems.

We first look at the performance of these codes for different network sizes. We consider regular topologies with various number of network coding stages between servers and clients with three nodes per coding stage. The source packets are protected by $(231, 249)$ Raptor codes. For all the links, the

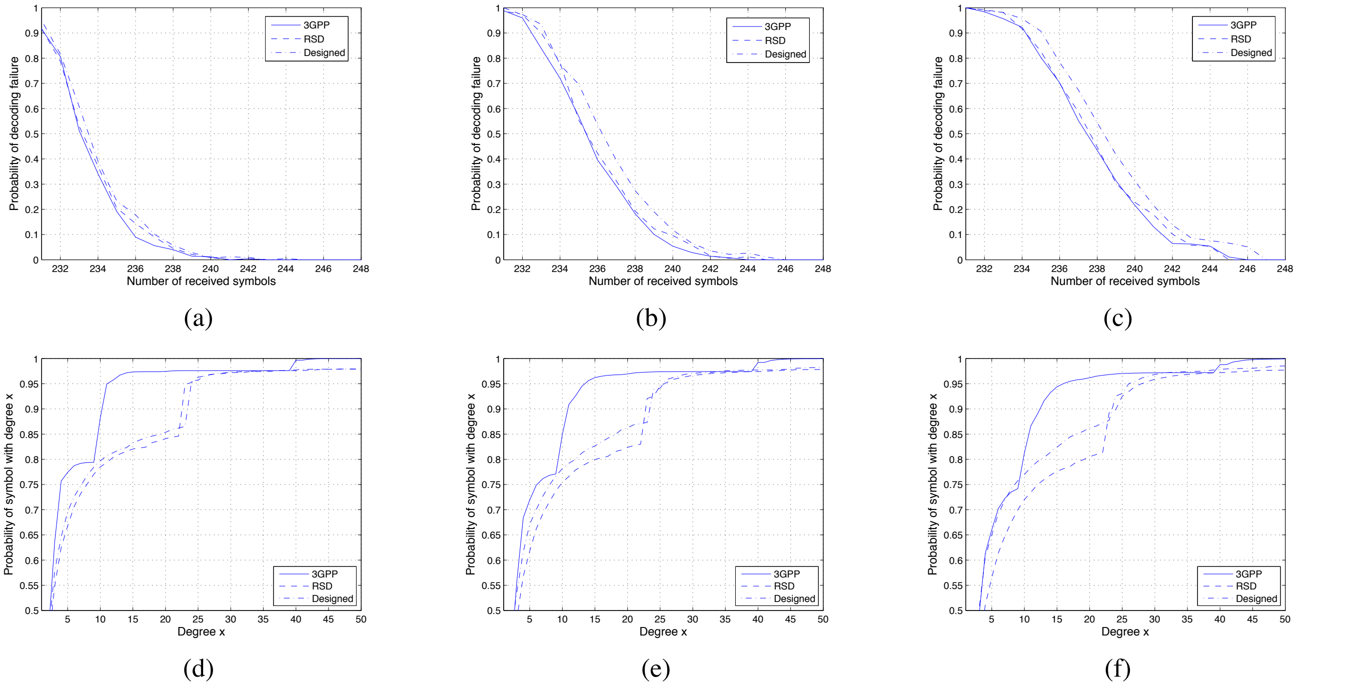

Fig. 2. Performance evaluation of $(231, 249)$ Raptor codes for regular network topologies with three intermediate nodes per stage for 3GPP codes (3GPP), 3GPP codes modified with RSD distribution (RSD) and the proposed distribution. The Raptor decoding failure probabilities at clients with respect to the number of received symbols are presented for (a) three, (b) six and (c) nine coding stages. Cumulative distribution of symbols degree received by each client are shown for networks with (d) three, (e) six and (f) nine coding stages.

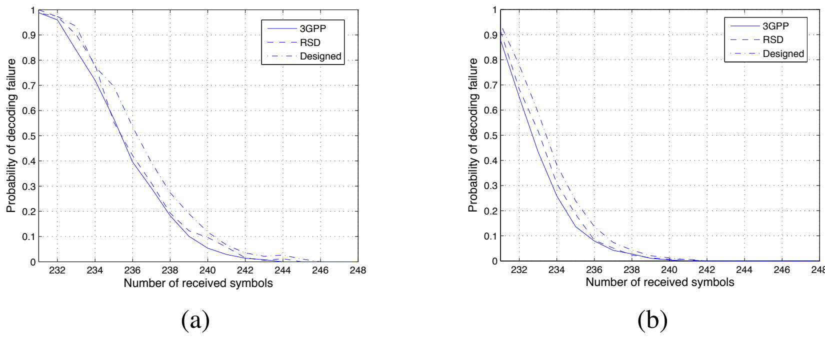

Fig. 3. Performance comparison of the network coding schemes employing with 3GPP, RSD and the proposed degree distribution functions at encoder, for regular networks with six stages and (a) three and (b) six nodes per coding stage.

symbol loss ratio is set to 5%, while the link bandwidth $\bar{b}$ permits the transmission of 83 symbols per link per time interval. The results for various network topologies are demonstrated in Fig. 2. The Raptor decoding probabilities with respect to the number of received packets for three-, six- and nine-stage network topologies are shown in Figs. 2(a), (b), and (c) respectively. For three-stage network topologies all methods perform equally well. As the number of stages increases, the proposed distribution performs slightly worst in terms of decoding probability, but it remains close to the performance of the other schemes. This performance degradation is due to the inefficiency of the design method in preserving the exact value of the spikes in the degree distribution function. This makes the proposed codes less efficient for large size topologies as the spike value becomes smaller than the ideal value. However, when the methods are compared in terms of the cumulative degree distribution function (cdf) as depicted in Figs. 2(d), (e), and (f), it is clear that the proposed codes outperform the 3GPP variant with RSD in terms of decoding complexity. The RSD scheme results in very dense generator matrices, while the designed code that targets a RSD-like degree distribution at the decoder results in sparser matrices and hence lower decoding complexity. This confirms that the proposed design algorithm succeeds in achieving low complexity decoding even after symbols have been combined by network coding.

Finally, note that both schemes have inferior performance to the 3GPP codes in terms of decoding probability. In addition, the 3GPP codes offers sparser generator matrices. This is expected as 3GPP codes optimize the pre-coder in order to use sparser LT generator matrices. However, we can observe in Figs. 2(d), (e), and (f) that the 3GPP codes result in denser matrices and become therefore less favorable to low-complexity decoding when the size of the network increases. The 3GPP codes have very sparse matrices, but network coding affects the sparsity of the generator matrices as the number of coding stages increases. For larger networks the cdf of the codes proposed in this paper becomes similar to the one of 3GPP codes for low degree values. They lead to similar decoding complexity as the 3GPP codes, which are among the most effective codes in practice. This shows that shaping the source (Raptor codes) degree distribution in order to guarantee low-complexity decoding is increasingly important for large network sizes.

We also study the performance of the different coding strategies for different number of nodes per coding stage. We consider network topologies with six coding stages, but respectively 3 and 6 network coding nodes per stage (see Figs. 3 (a) and (b)). We have set the link capacities in order to deliver 249 symbols to each receiver when transmission is error free. All links face 5% symbol loss rate. When network diversity is limited, the designed codes have inferior performance in terms of decoding probability, compared to that of the other two schemes. This performance difference is however not significant. All schemes take advantage of the improved network diversity as shown in Fig. 3(b). The performance gap between the designed codes and the RSD variant, however, decreases when the number of nodes per coding stage increases. In this case, the multiple reception probability[1] gets lower. This can be explained by a larger network diversity as more paths connect the clients with the servers, which favorably compensates for the larger probability of low degree symbols to be re-combined in network nodes. The optimized degree distribution permits to take advantage from the network diversity and leads to decoding performance that is competitive with the other distributions. At the same time, it guarantees a low decoding complexity, while the other source degree distributions lead to increased performance penalty when the network diversity augments.

V. Conclusions

In this paper, we presented a novel method for designing a degree distribution for Raptor like encoding at senders, so that the degree distribution of the received symbols could be controlled in network coding data delivery system. A generic optimization problem has been proposed for determining through geometric programming the appropriate source degree distributions in regular network topologies. The simulation results show that these proposed codes perform close to typical $3GPP$ Raptor codes in terms of decoding probability. At the same time, they permit to maintain a linear decoding time similar to ideal degree distribution functions such as RSD.

[1]The multiple reception probability is the probability that a node receive multiple times the same symbol, which is non-zero as symbols that have been combined can be combined again.

Acknowledgments

The authors would like to thank Dr Amin Shokrollahi for initial discussions on the degree optimization problem.

APPENDIX

Proof of lemma 1:

From eq. (6) for the spike at M we have $\Omega^1_M = c \cdot \Omega^1_{M-1}$ and thus we derive

$$\begin{aligned}\Omega_M &= c \cdot \Omega_{M-1} - \\ &- \frac{p}{(1-p)} \cdot \left\{ \sum_{j=1}^{M-1} \Omega_j \cdot \Omega_{M-j} c - \sum_{j=1}^{M-2} \Omega_j \cdot \Omega_{M-j-1} \right\}\end{aligned}$$

In order to determine the designing constraints for c, we examine when Ω_M takes negative values. Therefore, if $\Omega_M < 0$ it is

$$\begin{aligned}c \cdot \Omega_{M-1} < &- \frac{p}{(1-p)} \cdot c \cdot \sum_{j=1}^{M-2} \Omega_j \cdot \Omega_{M-j-1} + \\ &+ \frac{p}{(1-p)} \cdot \sum_{j=1}^{M-1} \Omega_j \cdot \Omega_{M-j}\end{aligned}$$

since $\Omega_{M-1} > 0$ and $0 \le p \le 1$ the inequality is true iff

$$\begin{aligned}&\Omega_1 \cdot \{c \cdot \Omega_{M-2} - 2 \cdot \Omega_{M-1}\} + \\ &+ \sum_{j=2}^{M-2} \Omega_j \cdot \{c \cdot \Omega_{M-j-1} - \Omega_{M-j}\} < 0\end{aligned}$$

We want all terms in the left side to be negative. Thus, $c < \frac{2 \cdot \Omega_{M-1}}{\Omega_{M-2}}$ and $c < \frac{\Omega_{M-j}}{\Omega_{M-j-1}}$, $j = 2, \ldots, M-2$.

As we have seen due to spike at M we have $\Omega^1_M = c' \cdot \Omega^1_{M+1}$. Similar to the above analysis it can be seen that $\Omega_M < 0$ when $c' < \frac{(1-p)}{2 \cdot \Omega_1 \cdot p}$ and $c' < \frac{\Omega_{M-j}}{\Omega_{M-j+1}}$, $j = 2, \ldots, M-1$.

References

[1] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network Information Flow," *IEEE Trans. Information Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[2] N. Thomos and P. Frossard, "Network Coding of Rateless Video in Streaming overlays," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 20, no. 12, pp. 1834–1847, Dec. 2010.

[3] A. Shokrollahi, "Raptor codes," *IEEE Trans. Information Theory*, vol. 52, no. 6, pp. 2551–2567, June 2006.

[4] M. Chiang, *Geometric Programming for Communications Systems*. Boston, MA: Now Publisers Inc, 2005.

[5] S. Puducheri, J. Kliewer, and T. E. Fuja, "The Design and Performance of Distributed LT codes," *IEEE Trans. Information Theory*, vol. 53, no. 10, pp. 3740–3754, Oct. 2007.

[6] M. Luby, "LT codes," in *Proc. of the 43rd Annual IEEE Symposium on Foundations of Computer Science (FOCS '02)*, Vancouver, Canada, Nov. 2002, pp. 271–280.

[7] "3GPP TS 26.346 V7.1.0, Technical Specification Group Services and System Aspects; Multimedia Broadcast/Multicast Service; Protocols and Codecs," June 2005.

[8] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge University Press, 2004.

[9] J. Lofberg, "Yalmip : A toolbox for modeling and optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004. [Online]. Available: http://control.ee.ethz.ch/ joloef/yalmip.php

[10] "The mosek optimization tools version 5.0. user's manual and reference," 2007, [Online]. Available: http://www.mosek.com.