On Distributed Distortion Optimization for Correlated Sources

Tao Cui, Tracey Ho, and Lijun Chen
Division of Engineering and Applied Science
California Institute of Technology
Pasadena, CA, USA 91125
Email: {taocui, tho}@caltech.edu, chen@cds.caltech.edu

***Abstract*— We consider lossy data compression in capacity-constrained networks with correlated sources. We develop, using dual decomposition, a distributed algorithm that maximizes an aggregate utility measure defined in terms of the distortion levels of the sources. No coordination among sources is required; each source adjusts its distortion level according to distortion prices fed back by the sinks. The algorithm is developed for the case of squared error distortion and high resolution coding where the rate distortion region is known, and is easily extended to consider achievable regions that can be expressed in a related form. Our distributed optimization framework applies to unicast and multicast with and without network coding. Numerical example shows relatively fast convergence, allowing the algorithm to be used in time-varying networks.**

I. Introduction

In this paper, we consider a network that has multiple correlated sources with associated distortion measures. In such a situation, we can integrate source coding and rate control by adapting the distortion of the sources to network congestion. Specifically, we consider adaptive lossy source coding for multicast with network coding [1], where each multicast session contains a set of continuous and possibly correlated sources.

For correlated sources, independent data compression is not an optimal strategy. Higher data compression efficiency can be obtained by using distributed source coding techniques. Existing approaches for network optimization with distributed source coding of correlated sources, e.g., [2], [3] for lossless coding and [4] for lossy coding, require coordination among the sources and do not admit fully distributed implementation.

Motivated by the optimization decomposition and utility maximization framework developed for TCP congestion control (see, e.g., [5], [6]), we consider the problem of maximizing an aggregate utility measure defined in terms of the distortion levels of the sources, e.g., minimum mean-square error (MMSE) distortion, and solve the problem to obtain a dual-based joint lossy source coding and network coding algorithm. The receiver-driven source coding algorithm adjusts distortion levels according to the distortion prices fed back from the sinks, and hence does not require coordination among the sources. With random network coding [7], our algorithm can be implemented in a fully distributed manner.

This work has been supported in part by DARPA grant N66001-06-C-2020, Caltech's Lee Center for Advanced Networking and a gift from Microsoft Research.

Our algorithm is developed for the case of squared error distortion and high resolution coding where the rate distortion region is known [8], and is easily extended to consider achievable regions that can be expressed in a related form. Our distributed optimization framework applies to unicast and multicast with and without network coding. Numerical examples show relatively fast convergence, allowing the algorithm to be used in time-varying networks.

II. Related Work

Joint optimization of source coding and routing/network coding for networks with correlated sources has been considered in a few recent works. In [2], joint optimization of lossless source coding and routing is proposed, where rate is allocated across sources to minimize a flow cost function under the constraint that the rates of the sources must lie in the Slepian-Wolf region. This approach is extended to lossy source coding in [4], where high-resolution source coding is assumed. A minimum cost subgraph construction algorithm for lossless source coding and network coding is proposed in [3], for the case of two sources.

Even though Slepian-Wolf coding is distributed, the optimization problems in [2]–[4] still require the coordination of the sources to guarantee that the source rates lie in the Slepian-Wolf region. Therefore, the algorithms in these works are not fully distributed.

In [9], [10], rate control for multicast with network coding has been studied for elastic traffic, with an aggregate utility maximization objective. The utility of each source is a function of its sending rate. In our work, the utility objective is defined in terms of distortion of each source. The rate distortion region imposes a new type of constraint on the optimization.

III. Preliminaries

A. Network and Coding Model

Consider a network, denoted by a graph $\mathcal{G}=(\mathcal{N},\mathcal{L})$, with a set $\mathcal{N}$ of nodes and a set $\mathcal{L}$ of directed links. We denote a link either by a single index l or by the directed pair (i,j) of nodes it connects. Each link l has a fixed finite capacity c_l packets per second. A set of multicast sessions $\mathcal{M}$ is transmitted over the network. Each session $m\in\mathcal{M}$ is associated with a set $\mathcal{S}_m\subset\mathcal{N}$ of sources and a set of $\mathcal{T}_m\subset\mathcal{N}$ of sinks. For session m, each source $s\in\mathcal{S}_m$ multicasts x^{ms} bits to all the sinks in $\mathcal{T}_m$. By flow conservation, we have, for any i, m, $s\in\mathcal{S}_m$ and $t\in\mathcal{T}_m$,

$$\sum_{j:(i,j)\in\mathcal{L}} g_{i,j}^{mst} - \sum_{j:(j,i)\in\mathcal{L}} g_{j,i}^{mst} = \begin{cases} x^{ms} & \text{if } i=s \\ -x^{ms} & \text{if } i=t \\ 0 & \text{otherwise} \end{cases}, \quad (1)$$

where $g_{i,j}^{mst}$ is the information flow rate over link (i,j) from source $s\in\mathcal{S}_m$ to sink $t\in\mathcal{T}_m$ for session m.

Network coding allows flows for different destinations of a multicast session to share network capacity by being coded together: for each multicast session m, with coding the actual physical flow on each link needs only be the maximum of the individual destinations' flows [1]. These constraints can be expressed as

$$g_{i,j}^{mst} \le f_{i,j}^{m}, \ \ (i,j)\in\mathcal{L}, \ m\in\mathcal{M}, \ (s,t)\in(\mathcal{S}_m,\mathcal{T}_m), \quad (2)$$

where $f_{i,j}^{m}$ gives the *physical flow* for session m. For the case of multiple sessions sharing a network, achieving optimal throughput requires coding across sessions. However, designing such codes is a complex and largely open problem. Thus, we simply assume that coding is done only across packets of the same session, i.e., intra-session coding. In this case, the set of feasible flow vectors is specified by combining constraints (1)-(2) for each session m with the link capacity constraint:

$$\sum_{m\in\mathcal{M}} f_{i,j}^{m} \le c_{i,j} \ ,\forall (i,j)\in\mathcal{L}. \quad (3)$$

In practice, the network codes can be designed using random linear network coding, see, e.g., [7], where for each node the data on outgoing links are random linear combination of the data on incoming links. If (1)-(2) holds, every sink can recover the transmitted packets with high probability. Please refer to [7] for a detailed description and discussion of overhead in random network coding and other practical implementation issues.

B. Lossy Source Coding

We consider multiterminal lossy source coding for continuous sources. Lossy source coding is data compression with a distortion measure. Wyner-Ziv coding [11] is a technique for distributed lossy source coding, for a single source with uncoded side information at the sink. The general distributed rate-distortion region for coding correlated sources in the general setting is unknown even for Gaussian sources [12] (Recently, the rate-distortion region for quadratic Gaussian two-terminal source-coding is found in [13]).

It is still an open problem whether or not in general the optimal solution can be separated into a simple quantization for each source followed by Slepian-Wolf lossless coding, but such separation exists in the high-resolution limit: the optimal rate-distortion performance can be achieved by separately quantizing each source, e.g. by dithered lattice quantizers, and then applying Slepian-Wolf lossless encoding to the quantizers' outputs. [8]. In the extreme of high resolution, it is shown in [8] that for squared-error distortion, the asymptotically achievable rate-distortion region for n correlated sources, $X_1,\ldots,X_n$, is given by

$$\sum_{X_i\in\mathcal{S}} R_i \ge h(\mathcal{S}|\mathcal{X}\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{X_i\in\mathcal{S}} D_i\right), \forall \mathcal{S}\subseteq\mathcal{X}, \quad (4)$$

where $\mathcal{X}=\{X_1,\ldots,X_n\}$, R_i and D_i are respectively the rate and MMSE distortion of X_i, and $h(\cdot)$ denotes differential entropy. A similar region is derived for more general difference distortion measures satisfying certain conditions. In general, the high-resolution region is an outer bound which becomes tighter as resolution increases. By using the results in [7], (4) can be readily extended to general networks by quantizing each source separately and then using random network coding.

For ease of exposition, we use the region defined in (4) in our subsequent development. Our results extend easily to the case where we have an achievable region of the form

$$\sum_{X_i\in\mathcal{S}} R_i \ge h(\mathcal{S}|\mathcal{X}\backslash\mathcal{S}) - \alpha_{\mathcal{S}}\log\prod_{X_i\in\mathcal{S}} D_i + \beta_{\mathcal{S}}, \forall\mathcal{S}\subseteq\mathcal{X}. \quad (5)$$

where $\alpha_{\mathcal{S}}$ and $\beta_{\mathcal{S}}$ are any constants. By appropriately choosing $\alpha_{\mathcal{S}}$ and $\beta_{\mathcal{S}}$, we can use (5) to approximate arbitrary achievable rate distortion region, e.g., that in [12].

IV. Distributed Algorithm

We assume for simplicity that each source transmits information over a single given multicast tree connecting it to its corresponding sink nodes. Such a multicast tree can be obtained by using protocols such as the distance vector multicast routing protocol [14]. These trees constitute an uncapacitated coding subgraph for each multicast session. Our distributed algorithm can be readily generalized to the case with multiple trees or without given trees (where the algorithm constructs coding subgraphs via back pressure) as in [10].

Let T^{ms} denote the multicast tree for source s in session m. Each tree T^{ms} contains a set $\mathcal{L}_{ms}\subseteq\mathcal{L}$ of links, which defines a $|\mathcal{L}|\times 1$ vector ξ^{ms} whose l-th entry is given by

$$\xi_l^{ms} = \begin{cases} 1 & \text{if } l\in T^{ms} \\ 0 & \text{otherwise.} \end{cases} \quad (6)$$

Similar to (2) and (3), with intra-session network coding we have the following two constraints

$$\xi_l^{ms} x^{ms} \le f_l^m, \qquad \forall l\in\mathcal{L}, m\in\mathcal{M}, s\in\mathcal{S}_m, \quad (7)$$

$$\sum_m f_l^m \le c_l, \qquad \forall l\in\mathcal{L}. \quad (8)$$

For lossy source coding, we assume that each source s of session m attains a utility $U_{ms}^D(D^{ms})$ when it compresses data at a distortion level D^{ms}, rather than a rate-dependent utility as in [5]. We assume that $U_{ms}^D(\cdot)$ is continuously differentiable, decreasing, and concave. This assumption enforces some kind of fairness among the sources, as the marginal utility is decreasing with the distortion. Examples of such utility functions are $\log(D_{\max}-D^{ms})$ and $-D^{ms}$, where $D_{\max}$ is the maximum tolerable distortion.

We formulate the source coding and network resource allocation problem as a utility maximization problem with the rate constraints (6)-(8) and the rate-distortion constraint (4) as follows.

$$\begin{aligned} \max_{D,x,f} \quad & \sum_{m\in\mathcal{M},s\in\mathcal{S}_m} U_{ms}^D(D^{ms}) \\ \text{s.t.} \quad & \xi_l^{ms} x^{ms} \le f_l^m, \forall l\in\mathcal{L}, m\in\mathcal{M}, s\in\mathcal{S}_m, \\ & \sum_{m\in\mathcal{M}} f_l^m \le c_l, \forall l\in\mathcal{L}, \\ & \sum_{s\in\mathcal{S}} x^{ms} \ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} D^{ms}\right), \forall\mathcal{S}\subseteq\mathcal{S}_m. \end{aligned} \quad (9)$$

Note that (9) is a convex problem and can be solved in polynomial time if all the utility and constraint information is given. However, a distributed algorithm is preferred in practice.

A. Algorithm

One way to derive a distributed solution is to consider its Lagrangian dual. However, in the rate-distortion constraint in (9), the source rates and distortions are not coupled at a single entity such as a node or link. We thus could not obtain a distributed algorithm by directly relaxing the rate-distortion constraint, which would still require source coordination. For the same reason, the algorithm in [4] is not fully distributed. In order to obtain a distributed solution, we consider the following equivalent problem

$$\begin{aligned}\max_{D,Z,x,y,f} \quad & \sum_{m,s} U_{ms}^D(D^{ms}) \\ \text{s.t.} \quad & \xi_l^{ms} x^{ms} \le f_l^m, \sum_m f_l^m \le c_l, \forall l,s, \\ & y^{mst} \le x^{ms}, Z^{mst} \le D^{ms}, \forall l,m,s,t, \\ & \sum_{s\in\mathcal{S}} y^{mst} \ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} Z^{mst}\right), \forall \mathcal{S}\subseteq\mathcal{S}_m,\end{aligned} \tag{10}$$

where, by introducing auxiliary variables y^{mst} and Z^{mst} at each sink $t\in\mathcal{T}_m$, we remove the troublesome coupling among the sources in the rate-distortion constraint. We will see later that these auxiliary variables admit physical interpretation and enable a distributed receiver-driven source coding algorithm.

Consider the Lagrangian dual to problem (10)

$$\min_{p\ge 0, q\ge 0, \lambda\ge 0} \phi(p,q,\lambda) \tag{11}$$

with partial dual function

$$\begin{aligned}\phi(p,q,\lambda) = & \max \sum_{m,s} U_{ms}^D(D^{ms}) - \sum_{l,m,s} p_l^{ms}(\xi_l^{ms} x^{ms} - f_l^m) \\ & - \sum_{m,s,t} q_t^{ms}(y^{mst} - x^{ms}) - \sum_{m,s,t} \lambda_t^{ms}(Z^{mst} - D^{ms}) \\ \text{s.t.} \quad & \sum_{m\in\mathcal{M}} f_l^m \le c_l, \\ & \sum_{s\in\mathcal{S}} y^{mst} \ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} Z^{mst}\right),\end{aligned} \tag{12}$$

where we relax only the first and the third constraints in (10) by introducing Lagrange multiplier p_l^{ms} at link l for source s in session m, and q_t^{ms} and λ_t^{ms} at sink t for source s in session m. The dual function $\phi(p,q,\lambda)$ has a nice decomposition structure into four separate subproblems

$$\begin{aligned}\phi_1(q,\lambda) = & \min_{y,Z} \sum_{m,s,t} q_t^{ms} y^{mst} + \lambda_t^{ms} Z^{mst}, \\ & \text{s.t.} \sum_{s\in\mathcal{S}} y^{mst} \ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} Z^{mst}\right),\end{aligned} \tag{13}$$

$$\phi_2(\lambda) = \max_D \sum_{m,s} U_{ms}^D(D^{ms}) + \sum_{m,s}\left(\sum_t \lambda_t^{ms}\right) D^{ms}, \tag{14}$$

$$\phi_3(p,q) = \min_x \sum_{m,s} x^{ms}\left(\sum_l p_l^{ms}\xi_l^{ms} - \sum_t q_t^{ms}\right), \tag{15}$$

$$\phi_4(p) = \max_f \sum_{l,m,s} p_l^{ms} f_l^m, \text{ s.t.} \sum_{m\in\mathcal{M}} f_l^m \le c_l. \tag{16}$$

The first subproblem is the minimum weighted rate and distortion problem for virtual lossy source coding at each sink. This has same similarity with the reverse backpressure algorithm in [15] for distributed control of lossless source coding. The second subproblem is distortion control. The third one is rate allocation. The fourth one is joint network coding and session scheduling. Thus, by dual decomposition, the problem decomposes into separate "local" optimization problems of application, transport, and network/link layers, respectively. The four problems interact through dual variables p, q, λ.

Lossy source coding: The virtual joint rate allocation and data compression problem (13) can further decompose into separate optimization problems at each sink $t\in\mathcal{T}_m$,

$$\begin{aligned}\min_{y,Z} \; & \sum_s q_t^{ms} y^{mst} + \lambda_t^{ms} Z^{mst} \\ \text{s.t.} \; & \sum_{s\in\mathcal{S}} y^{mst} \ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S}) - \log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} Z^{mst}\right).\end{aligned} \tag{17}$$

For fixed Z^{mst}, it can be readily verified that the polyhedron described by the constraint in (17) is a contra-polymatroid [16]. From Lemma 3.3 in [16], a greedy algorithm solves (17) optimally. Let π^* be any permutation of $\mathcal{S}_m$ such that $q_t^{m\pi^*(1)} \le q_t^{m\pi^*(2)} \le \cdots \le q_t^{m\pi^*(|\mathcal{T}_m|)}$. Then, by Lemma 3.3 in [16], the solution of (17) with given Z is given by

$$\begin{aligned} & y^{m\pi^*(1)t}(q) = h(\pi^*(1)) - \log\left(2\pi e Z^{m\pi^*(1)t}\right), \\ & y^{m\pi^*(2)t}(q) = h(\pi^*(2)|\pi^*(1)) - \log\left(2\pi e Z^{m\pi^*(2)t}\right), \\ & \cdots \\ & y^{m\pi^*(|\mathcal{T}_m|)t}(q) = h(\pi^*(|\mathcal{T}_m|)|\pi^*(|\mathcal{T}_m|-1),\ldots,\pi^*(1)) \\ & \qquad - \log\left(2\pi e Z^{m\pi^*(|\mathcal{T}_m|)t}\right).\end{aligned} \tag{18}$$

Substituting (18) into (17) and minimizing (17) over Z^{mst}, we get

$$Z^{mst}(q,\lambda) = \frac{q_t^{ms}}{\lambda_t^{ms}}. \tag{19}$$

Substituting (19) into (18), we obtain the optimal $y^{mst}(q,\lambda)$.

Now, consider the distortion control problem (14). At source s, at each time slot τ, instead of solving (14) directly for D^{ms}, we update D^{ms} using a primal subgradient algorithm according to

$$D^{ms}(\tau+1) = \left[D^{ms}(\tau) + \epsilon_\tau\left(U_{ms}^{D\,\prime}(D^{ms}(\tau)) + \sum_t \lambda_t^{ms}\right)\right]^+, \tag{20}$$

where ϵ_τ is a positive scalar stepsize, and $^+$ denotes the projection on the set of non-negative real numbers. We will see that λ_t^{ms} can be interpreted as the price resulting from the mismatch between the source distortion and virtual source distortion at the sink. The source distortion is adjusted according to the aggregate *distortion price* $\sum_t \lambda_t^{ms}$ due to virtual source coding, which is fed back from the sinks of session m.

Rate allocation: To recover the source rate, instead of solving (15) directly, we update the source rate using a primal subgradient algorithm. At time $\tau+1$, the source rate $x^{ms}(\tau+1)$ is updated according to

$$x^{ms}(\tau+1) = \left[x^{ms}(\tau) - \epsilon_\tau\left(\sum_l p_l^{ms}\xi_l^{ms} - \sum_t q_t^{ms}\right)\right]^+. \tag{21}$$

Each source then compresses data according to rate $x^{ms}(\tau+1)$ by using dithered lattice quantizers [8] and randomized linear network coding.

Session scheduling and network coding: For each link l, find the session $m_l^*=\text{argmax}_m \sum_s p_l^{ms}$. A random linear combination of packets from all the sources in session m_l^* is sent at the rate of c_l. This is equivalent to solving (16) by the following assignment

$$f_l^m(p)=\begin{cases} c_l & \text{if } m=m_l^* \\ 0 & \text{otherwise.} \end{cases} \tag{22}$$

Dual variable update: By using the first order Lagrangian method [17], at time $\tau+1$, the dual variables are updated according to

$$p_l^{ms}(\tau+1)=[p_l^{ms}(\tau)+\gamma_\tau(\xi_l^{ms}x^{ms}(p(\tau),q(\tau))-f_l^m(p(\tau)))]^+_; \tag{23}$$

$$q_t^{ms}(\tau+1)=\left[q_t^{ms}(\tau)+\gamma_\tau\left(y^{mst}(q(\tau),\lambda(\tau))-x^{ms}(p(\tau),q(\tau))\right)\right]^+_; \tag{24}$$

$$\lambda_t^{ms}(\tau+1)=\left[\lambda_t^{ms}(\tau)+\gamma_\tau\left(Z^{mst}(q(\tau),\lambda(\tau))-D^{ms}(\lambda(\tau))\right)\right]^+_; \tag{25}$$

where γ_τ is a positive scalar stepsize. Note that (23)-(25) are distributed and can be implemented by individual links and sinks using only local information. The algorithm (18)-(25) is a distributed primal-dual subgradient algorithm for problem (10) and its dual. By using Lyapunov method and extending the techniques for the dual subgradient method as in, e.g., [10], we can prove that the algorithm (18)-(25) converges to within an arbitrarily small neighborhood of the optimal solution of (10) by using suitable stepsizes $\epsilon_\tau,\gamma_\tau$.

Note that p_l^{ms} results from the rate constraint and thus can be interpreted as a virtual congestion price at the link. q_t^{ms} can be interpreted as the price resulting from the mismatch between the physical source rate and virtual source rate at the sink, and λ_t^{ms} as the price resulting from the mismatch between the source distortion and virtual source distortion at the sink. Our adaptive source coding is a receiver-driven scheme. Since the sink receives information from all the sources, it can estimate the rate-distortion region of correlated sources, and solve a virtual joint rate allocation and data compression problem. By adapting to the prices q_t^{ms} and λ_t^{ms}, the source tries to match the virtual rate and distortion. The source rate also adapts to p_l^{ms} to avoid congestion.

Note that in our algorithm the sink does not feedback any information about the source distributions to the sources. Feeding back this information may change the rate-distortion region and improve the solution, but this is beyond the scope of this paper.

B. Numerical Example

In this subsection, we provide numerical examples to complement the analysis in previous sections. We consider a simple network as shown in Fig.1.(a). For simplicity, we assume that there is only one multicast session with two correlated sources s_1 and s_2, and two sinks t_1 and t_2. The capacity of link $(s_1,1)$ is 0.4 and the capacity of link $(s_2,1)$ is 0.3. All the other links have unit capacity. We assume that all the sources have the same utility function $U^D(D)=\log(1-D)$. We also assume that $h(s_1|s_2)=h(s_2|s_1)=0.2$ and $h(s_1)=h(s_2)=0.5$.

The multicast tree for source s_1 is chosen as $\{(s_1,1),(1,2),(2,t_2),(s_1,t_1)\}$, and for source s_2 is chosen as $\{(s_2,1),(1,2),(2,t_1),(s_2,t_2)\}$. Fig.1.(b)-(c) show the evolution of source distortions and source rates versus the number of iterations for lossy source coding with stepsizes $\epsilon=1$ and $\gamma=0.01$. We see that both source distortions and rates converge quickly to a neighborhood of the corresponding optimum and oscillate around them.

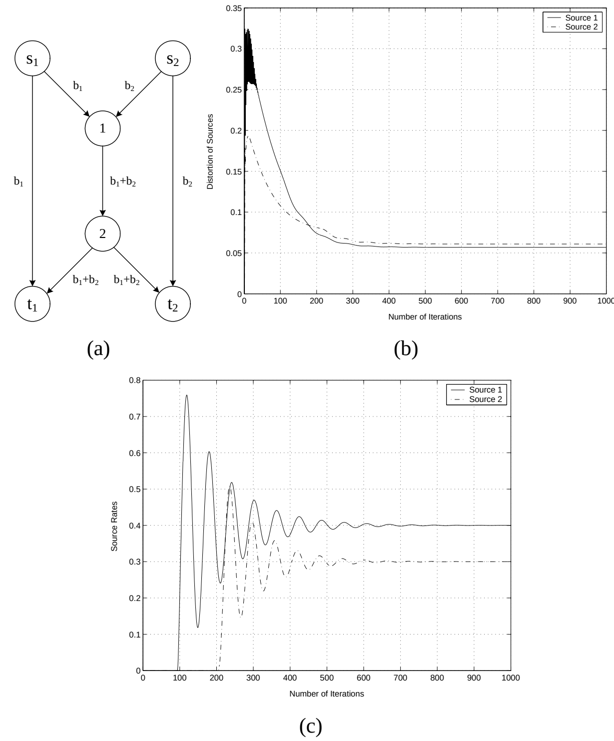

Fig. 1. (a) The butterfly network; (b) The evolution of source distortions versus the number of iterations; (c) The evolution of source rates versus the number of iterations.

V. Extensions and Practical Considerations

1) Extension to the networks without given multicast trees: We has been focused on the networks with given multicast trees. The same solution approach applies to the networks without given multicast trees. Since no multicast tree is given, we will use the rate constraint (1)-(3) instead of (7)-(8), and solve the following utility maximization problem

$$\begin{aligned} \max \quad & \sum_{m\in\mathcal{M},s\in\mathcal{S}_m} U_{ms}^D(D^{ms}) \\ \text{s.t.} \quad & \sum_{j:(i,j)\in\mathcal{L}} g_{i,j}^{mst}-\sum_{j:(j,i)\in\mathcal{L}} g_{j,i}^{mst}=\begin{cases} x^{ms} & \text{if } i=s \\ -x^{ms} & \text{if } i=t \\ 0 & \text{otherwise,} \end{cases} \\ & g_{i,j}^{mst}\le f_{i,j}^m, \sum_m f_{i,j}^m\le c_{i,j}, \forall(i,j)\in\mathcal{L},m,s,t, \\ & \sum_{s\in\mathcal{S}} x^{ms}\ge h(\mathcal{S}|\mathcal{S}_m\backslash\mathcal{S})-\log\left((2\pi e)^{|\mathcal{S}|}\prod_{s\in\mathcal{S}} D^{ms}\right), \forall\mathcal{S}\subseteq\mathcal{S}_m, \end{aligned} \tag{26}$$

We can apply the same approach as in Section IV to solve the above problem, and obtain a distributed joint source coding, rate allocation, and network coding/session scheduling algorithm. This resulting algorithm works in a similar way as the algorithm (18)-(25), but with some minor but subtle differences. For example, the session scheduling component will use back-pressure to do optimal scheduling, similarly to [15]. We will not elaborate on these, due to the space limit.

2) Multicast without Network Coding: In current network, multicast is dominated by routing based method without network coding. Our joint rate allocation and source coding framework can also be applied to this case. Mathematically, network coding comes into action through rate constraint (7). In routing based multicast, (7) is replaced by $\sum_s \xi_l^{ms} x^{ms} \leq f_l^m$. It is straightforward to carry out joint rate allocation and source coding in the same way as in section IV, with only a slight modification. The case of multiple unicasts can also be included in our framework, as unicast can be seen as a special case of multicast.

3) Practical Source Codes and Network Codes: We have assumed the use of random linear network codes and minimum entropy decoding, which has a high decoding complexity. To reduce the complexity, we can separate source coding and network coding. Random [7] or deterministic [18] network codes can be used for network coding. We can concatenate dithered lattice quantizers to the LDPC based Slepian-Wolf encoders [19]. It has been shown in [20] that in many cases, separated network coding and lossless source coding preserves optimality. We expect that separated network coding and lossy source coding is also optimal in many cases, at least in the high-resolution case where quantization can be separated from lossless source coding.

4) Layered Source Coding: In heterogeneous networks, encoded bitstreams may be characterized by a hierarchy of importance layers [21]. Each layer corresponds to a multicast session. In [21], sinks adjust their reception rate by simply joining and leaving multicast sessions. Note that our proposed algorithm is for multiple session multicast. We can also adapt our algorithm to this layered source coding framework. Slepian-Wolf coding can be applied to sources in the same layer. Each sink subscribes to only one layer at first. As the congestion price is an indication of network congestion, if a sink observes that the congestion price converges, it subscribes to another multicast session corresponding to a higher layer. If the congestion price does not converge, the sink drops a layer.

5) Entropy and Probability Density Estimation: State-of-the-art distributed source codes need the knowledge of joint probability density function (pdf) of all the sources in each session for both encoding and decoding. It is hard for all the sources to learn this information. Our proposed framework relaxes this constraint by requiring that only sinks need it. A possible approach for estimating the joint pdf at the sinks is for the sources to initially transmit quantized data without Slepian-Wolf coding. On receiving this data, the sinks estimate the joint pdf by using well-developed techniques in multivariate density estimation.Later, the estimated pdf can be refined by the decompressed data. Cyclic redundancy checks can be used to detect errors in the decoded data, in which case the rate-distortion region is conservatively modified such that the next data frame can be decoded correctly. However, as pdf estimation is complicated, it is desirable to have universal distributed source codes.

VI. Conclusion

We have presented a fully distributed algorithm for adaptive lossy source coding for multicast with network coding, where each session contains a set of correlated sources. Based on the utility maximization framework and its decomposition, we proposed a distributed algorithm for joint optimization of source coding and network coding. The resulting receiver-driven algorithm adjusts distortion levels according to distortion prices fed back from the sinks, and hence does not require coordination among the sources. With random network coding, the algorithm can be implemented in a fully distributed manner. In this work we have used the known rate distortion region for high resolution lossy source coding; our work easily extends to achievable regions that can be expressed in a related form. It would be interesting to extend our work to other achievable rate distortion regions.

References

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

[2] R. Cristescu, B. Beferull-Lozano, and M. Vetterli, "Networked Slepian-Wolf: Theory, algorithms and scaling laws," *IEEE Trans. Inform. Theory*, vol. 51, no. 12, pp. 4057–4073, Dec. 2005.

[3] A. Lee, M. Médard, K. Haigh, S. Gowan, and P. Rubel, "Minimum-cost subgraphs for joint distributed source and network coding," in *Proc. of Workshop on Network Coding, Theory and Applications*, Jan. 2007.

[4] R. Cristescu and B. Beferull-Lozano, "Lossy network correlated data gathering with high-resolution coding," *IEEE/ACM Trans. Networking*, vol. 14, pp. 2817–2824, June 2006.

[5] F. Kelly, A. Maulloo, and D. Tan, "Rate control in communication networks: shadow prices, proportional fairness and stability," *J. Oper. Res. Soc.*, vol. 49, no. 3, pp. 237–252, March 1998.

[6] M. Chiang, S. H. Low, A. R. Calderbank, and J. C. Doyle, "Layering as optimization decomposition," to appear in *Proc. of IEEE*, Jan. 2007.

[7] T. Ho, R. Koetter, M. Médard, M. Effros, J. Shi, and D. Karger, "A random linear network coding approach to multicast," *IEEE Trans. Inform. Theory*, vol. 52, no. 10, pp. 4413–4430, Oct. 2006.

[8] R. Zamir and T. Berger, "Multiterminal source coding with high resolution," *IEEE Trans. Inform. Theory*, vol. 45, pp. 106–117, Jan. 1999.

[9] Y. Wu and S.-Y. Kung, "Distributed utility maximization for network coding based multicasting: A shortest path approach," *IEEE J. Select. Areas Commun.*, vol. 24, no. 8, pp. 1475–1488, Aug. 2006.

[10] L. Chen, T. Ho, S. H. Low, M. Chiang, and J. C. Doyle, "Rate control for multicast with network coding," to appear in *IEEE Infocom*, 2007.

[11] A. D. Wyner and J. Ziv, "The rate-distortion function for source coding with side information at the decoder," *IEEE Trans. Inform. Theory*, vol. 22, no. 1, pp. 1–10, Jan. 1976.

[12] T. Berger, "Multiterminal source coding," in *The Information Theory Approach to Communications*, G. Longo, Ed. Springer-Verlag, 1977.

[13] A. Wagner, S. Tavildar, and P. Viswanath, "The rate region of the quadratic gaussian two-terminal source-coding problem," arXiv:cs.IT/0510095.

[14] D.Waitzman, S. Deering, and C. Partridge, "Distance vector multicast routing protocol," RFC 1075, Nov.1988.

[15] T. Ho and H. Viswanathan, "Dynamic algorithms for multicast with intra-session network coding," in *Proc. of Allerton Conference on Communication, Control, and Computing*, Sept. 2005.

[16] D. Tse and S. Hanly, "Multiaccess fading channels-part I: Polymatroid structure, optimal resource allocation and throughput capacities," *IEEE Trans. Inform. Theory*, vol. 44, no. 7, pp. 2796–2815, Nov. 1998.

[17] D. P. Bertsekas, *Nonlinear Programming*. Athena Scientific, 1995.

[18] S. Jaggi, P. Sanders, P. Chou, M. Effros, S. Egner, K. Jain, and L. Tolhuizen, "Polynomial time algorithms for multicast network code construction," *IEEE Trans. Inform. Theory*, vol. 51, no. 6, pp. 1973–1982, June 2005.

[19] A. D. Liveris, Z. Xiong, and C. N. Georghiades, "Compression of binary sources with side information at the decoder using LDPC codes," *IEEE Commun. Lett.*, vol. 6, pp. 440–442, Oct. 2002.

[20] A. Ramamoorthy, K. Jain, P. Chou, and M. Effros, "Separating distributed source coding from network coding," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2785–2795, June 2006.

[21] S. McCanne, M. Vetterli, and V. Jacobson, "Low-complexity video coding for receiver-driven layered multicast," *IEEE J. Select. Areas Commun.*, vol. 15, no. 6, pp. 983–1001, Aug. 1997.