

# Joint Source and Flow Optimization for Scalable Video Multirate Multicast Over Hybrid Wired/Wireless Coded Networks

Chenglin Li, Hongkai Xiong, *Senior Member, IEEE*, Junni Zou, *Member, IEEE*, and Zhihai He, *Senior Member, IEEE*

**Abstract**—This paper aims to optimize the overall video quality and traffic performance for multi-rate video multicast over hybrid wired/wireless networks. In order to perform layered utility maximization over tiered networks, we propose a joint source-network flow optimization scheme where individual layers of the scalable video stream are imposed on their optimal multicast paths and associated rates for the highest sustainable layered video quality with minimum costs. It sufficiently guarantees that each destination node accesses progressive layered stream in an incremental order, considers network coding across overlapping paths to destination nodes for decent multicast capacity, and addresses the link contention problem during wireless transmission. We formulate the problem into convex programming with the objective to minimize the total rate-distortion variations between layers. Using primal decomposition and the primal-dual approach, we develop a decentralized algorithm with two levels of optimization. The numerical and packet-level results compare extensive performance under different control conditions over coded and non-coded hybrid networks. It demonstrates that the proposed algorithm could actually achieve the max-flow throughput and provide better video quality with optimal layered access for heterogeneous receivers.

**Index Terms**—Multi-rate multicast, network coding, rate-distortion, resource allocation, scalable video coding.

## I. INTRODUCTION

MULTI-RATE MULTICAST has emerged as an important method for content distribution over large networks with its capability in adapting to different user requirements and time-varying network conditions [1]. From a source coding perspective, scalable video coding (SVC) allows rate adaptation not only at the encoder/decoder, but also at intermediate network nodes while achieving highly efficient

rate-distortion performance [2]. An SVC stream consists of a base layer and one or multiple enhancement layers with a flexible multi-dimensional layered structure, providing various operating points in spatial resolutions, temporal frame rates, and reconstruction quality levels. Different SVC layers with multi-rate multicast are transported in different IP multicast groups which are subscribed by heterogeneous receivers with different computation and communication resources and capabilities. Within this context, layered multi-rate multicasting is equivalent to a generalized multi-source problem where the progressive inter-layer dependency is considered as fairness between different sources [9], [11].

Recently, the hybrid wired/wireless network has become an increasingly important tool for communication [3]–[5], which is formed by placing a sparse network of base stations in an ad hoc network, where base stations are connected by a high-bandwidth wired network and act as relays for wireless nodes. Hybrid networks present a tradeoff between traditional wired networks and pure ad hoc networks. By introducing the flexibility and scalability that traditional wired-only networks cannot achieve, data may be forwarded in a wireless manner or through the high capacity wired links. On the other hand, the addition of wired base station nodes in hybrid networks is a natural approach to reducing the energy and traffic burden on ad hoc nodes and increasing the system throughput. Furthermore, hybrid ad hoc network model can be viewed as a means to extend the communication coverage of wireless cellular infrastructure. Due to the heterogeneous network performance where different wireless users are capable of distinct access capacities and correspondingly require different quality of service, the hierarchical hybrid wired/wireless network structure can benefit the asymptotic capacity in terms of flexibility and scalability. Such advantages make the multicast over hybrid networks suitable for multimedia dissemination with different quality levels. In this paper, we aim to develop an efficient flow control and performance optimization scheme for scalable video streaming over hybrid wired/wireless networks. Fig. 1 shows an example, where the layered SVC stream is generated at the source node, and distributed to different users through the two-tier wired and wireless network. The objective is to maximize the overall video quality of all receivers.

The contribution of this paper is twofold. First, we formulate a mathematically rigorous convex optimization problem with

Manuscript received May 5, 2010; revised September 21, 2010 and November 6, 2010; accepted November 9, 2010. Date of publication March 17, 2011; date of current version May 4, 2011. This work was supported in part by the NSFC, under Grants 60632040, 60772099, 60802019, and 60928003, and by the Program for New Century Excellent Talents in University (NCET-09-0554). This paper was recommended by Associate Editor P. Frossard.

C. Li and H. Xiong are with the Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: lcl1985@sjtu.edu.cn; xionghongkai@sjtu.edu.cn).

J. Zou is with the School of Communication and Information Engineering, Shanghai University, Shanghai 200072, China (e-mail: zoujn@shu.edu.cn).

Z. He is with the Department of Electrical and Computer Engineering, University of Missouri, Columbia, MO 65211 USA (e-mail: hezhi@missouri.edu).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TCSVT.2011.2129070

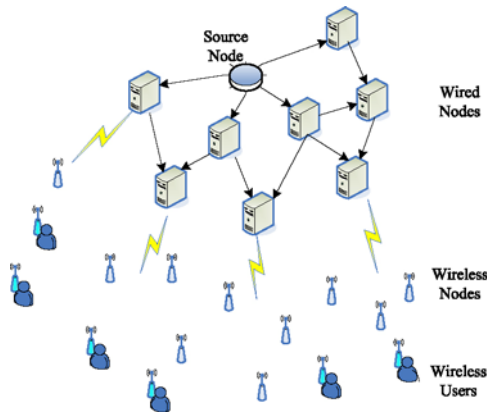


Fig. 1. Illustration of a two-tier wired/wireless network.

the objective to minimize the total quality variation among layers, which considers the inter-layer dependency constraints of SVC bit streams to achieve the highest sustainable layered video quality with minimum costs. The advantages of this optimization formulation include: it guarantees that each destination node accesses progressive layered stream in an incremental order; network coding is utilized for a decent multicast capacity; the link contention of wireless transmission is addressed in the formulation; and a partial quality layer is allowed to be received by introducing a tolerable rate region for each SVC layer. Second, we develop a distributed algorithm with two levels of optimization to maximize the total network utility by jointly optimizing the allocated source rates, transmission flow rates, and routing scheme. The proposed distributed algorithm is proven to be asymptotically stable using the Lyapunov theorem. The extensive numerical and packet-level results demonstrate that the proposed algorithm could actually achieve the max-flow throughput and benefit overall received video quality.

The rest of this paper is organized as follows. Related work is discussed in Section II. The notations and network models are described in Section III. In Section IV, we formulate the flow control and resource allocation problem for scalable video coding and multi-rate multicast over hybrid wired/wireless networks. The decentralized algorithm for SVC-based multi-rate multicasting with network coding based routing is proposed in Section V. We also prove the stability of the proposed decentralized rate control scheme and provide an efficient implementation scheme. Numerical experimental results and packet-level simulations are presented in Section VI. Section VII concludes this paper.

## II. RELATED WORK

In communication networks, to achieve the capacity in single-source multiple-terminal multicast [6], network coding is often performed at intermediate nodes. Recent research has demonstrated that it is able to significantly improve the network throughput and robustness to link or node failure and packet losses [7]. Distributed random linear coding schemes [7], [17] have been proposed for practical implementation of network coding. Chen *et al.* [8] developed adaptive rate control algorithms for networks with and without coding sub-graphs.

In this paper, we incorporate network coding into scalable video streaming over the network to optimize the overall content distribution performance.

Several rate control schemes have been developed in the literature on scalable video coding, layered multicast, and network coding [9]–[12]. For example, the distributed rate allocation scheme in [9] addressed the problem of rate allocation for video multicast with SVC over wireless mesh networks, with the goal of minimizing total video distortion of all peers. Yan *et al.* [10] used a rate-distortion function as the application utility measure for optimizing the overall video quality. These rate control methods improve the performance of scalable video streaming over hybrid networks. However, they are suboptimal solutions and the inter-dependency between SVC layers has not been adequately addressed. This might cause decoding problem in scalable video streaming over networks. For example, video packets from higher-layers may arrive before or without packets from lower layers, which will cause decoding failure. Recently, network coding has emerged as an important tool for maximizing the network throughput during content distribution. Incorporating network coding to flow control and resource allocation allows us to achieve the maximum network throughput and performance limit. A recent work by Zhao *et al.*, called the LION algorithm [11], demonstrated that incorporating both inter-layer dependency and network coding into flow control is able to significantly improve the network throughput for overlay multicast. However, it should be noted that the LION algorithm is only a heuristic method that progressively organizes receivers into separated layered meshes and achieves only sub-optimal performance. In addition, it fails to provide rigorous theoretical justifications on the relation between the LION algorithm and the initial layered overlay multicast problem. As a further improvement, [12] proposed a prioritized flow optimization formulation for SVC and multicast over heterogeneous networks, which used the path cost and prices of each layer as the priority parameters to capture inter-layer dependency and provided rigorous distributed algorithms proven to be stable and convergent.

## III. NOTATIONS AND NETWORK MODELS

We consider video content distribution over a tiered wired and wireless network. The network can be modeled as a directed graph  $G_1 \cup G_2$ , where  $G_1 = (V_1, E_1)$  denotes the wired network and  $G_2 = (V_2, E_2)$  represents the wireless network. In the wired network  $G_1$ ,  $E_1$  is the set of wired links and  $V_1 = \{s\} \cup N \cup T$  is the set of wired nodes, where  $\{s\}$ ,  $N$  and  $T$  represent the set of source nodes, relay nodes and receiver nodes, respectively. The wireless network  $G_2$  is composed of the set of wireless links  $E_2$  and the set of wireless nodes  $V_2 = T \cup R \cup D$ , where  $R$  and  $D$  denote the relay nodes and destination nodes, respectively, and  $T$  represents the set of source nodes in  $G_2$  which is also the set of receiver nodes in  $G_1$ . Hereinafter, we write  $V = \{s\} \cup N \cup T \cup R \cup D$  and  $E = E_1 \cup E_2$ . Let each wired link  $l \in E_1$  have a finite capacity of  $C_l$ , and consider the wireless link contention in a shared transmission medium of wireless link  $l \in E_2$  with capacity  $C$  [20].

### A. SVC Coding Model

Utilizing SVC technique, a scalable bit stream could be represented in two different ways, as a layered representation (layered scalable) or a flexible combined scalability (fully scalable) [34]. Generally, the flexible combined scalability can benefit the scenario of unicast, where the target stream can be extracted at any bit rate from the SVC elementary stream in compliance with the single receiver's capacity status. In comparison, the layered scalability can benefit the network multicasting by offering simple adaptation operation to heterogeneous receivers, i.e., different receivers can subscribe to different combinations of layers under the constraints of network capacity and layer dependency.

Therefore, we adopt the layered scalability for its simplicity and assume that the SVC video stream is encoded into a set of  $M$  layers  $\{L_1, L_2, \dots, L_M\}$  with a predefined encoding rate based on the network condition. According to the encoding rates of layers, we can make the optimal adaptation decision in the scalability cube model illustrated in Fig. 2 by mapping from an SVC elementary stream with fully scalable representation into the layered representation. Correspondingly, the multicast of SVC video stream is divided into  $M$  multicast sessions. Each multicast session  $m$  has one source node  $s$ , a set of destination nodes  $D$ , and a set of relay nodes  $NUTUR$ . In order to successfully decode received SVC video streams, we should make sure that all destination nodes are able to subscribe to SVC multicast layers in an incremental order, since layer  $m+1$  is not decodable without its previous layers 1 to  $m$ . This layer dependency constraint promises the most efficient SVC decoding at each destination node.

In fact, a practical SVC encoder and decoder during fluctuating network adaptation could perform a very large variation of options to make a flexible inter-layer motion-compensated prediction and reconstruction [38]. It is well known that the standardized scalable extension of H.264 (SVC) only specifies the bit-stream syntax and bounds to facilitate different applications. To be concrete, a higher layer might be able to be decoded even if the lower dependent layers are either truncated or partially dropped to a mild extent. Certainly, it would cause a drift between the decoded pictures reconstructed in the encoder and in the decoder. To prevent error propagation, the reconstructed reference pictures in the motion-compensated prediction would be limited in the highest layer which is completely received. Under the condition, non-normative coding techniques, e.g., a variety of error resilience and error concealment tools, would be utilized to estimate the lost information for the decoding of the higher layer. It makes a receiver possible to subscribe to a partial quality layer within an achievable rate region. Considering both fluctuating network adaptation and optimization condition [39], [40], each layer is distributed over a multicast session at a variable transmission rate within a tolerable rate region  $[b_m, B_m]$ . Mathematically, the upper bound  $B_m$  (e.g., the encoding rate with a resilient margin) and the lower bound  $b_m$  (e.g., the minimum partial margin for layer  $m$ ) are specified for a confidence interval of the layered transmission rate in layer  $m$ . It differentiates the layers with the piecewise confidence intervals along the layer-dependent direction, namely, the

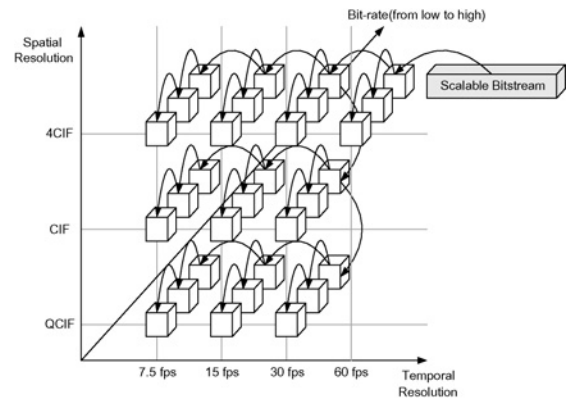


Fig. 2. Typical structure of scalable video bitstream with multiple dimensions.

achievable transmission rate for each layer is mathematically extended from an encoding rate point to a tolerable rate region. From the layered optimization, the fine-granular continuity of the targeted variables (rates) could specifically urge the convexity of optimization problem for developing a distributed solution, and support a strong notion of fairness.

### B. Network Coding Model

The algebraic operation in packet level at intermediate nodes, called *network coding* [6], recently has attracted significant research interests [8], [13]. Besides improving communication network's throughput, various potential benefits of network coding have been found, including robustness to link/node failures [14] and packet losses [15], [16]. Distributed random linear coding schemes [7], [17], on the other hand, have made practical implementation of network coding possible. To transmit multiple multicast sessions over a shared network, we might perform network coding across sessions to achieve the optimal throughput. However, combining data belonging to different layers makes it difficult to recover all original data for destination nodes that only receive partial layers. Thus, we limit network coding within each session in this paper. This approach is often referred to as intra-session coding or superposition coding [18].

With intra-session network coding, flows to different destinations of a multicast session are allowed to share network capacity by being coded together. For a single multicast session  $m$  with transmission rate  $R_m \in [b_m, B_m]$ , information flow must be transmitted at rate  $R_m$  to each destination. However, with network coding, we only need to set the actual physical flow on each link to be the maximum of the individual destinations' information flows. Specifically, for link  $l = (i, j)$ , let  $x_{(i,j)}^{md}$  denote the information flow for destination  $d$  of multicast session  $m$ , and  $f_{(i,j)}^m$  denote the physical flow for multicast session  $m$ , then these constraints can be expressed as follows:

$$\sum_{j:(i,j) \in E} x_{(i,j)}^{md} - \sum_{j:(j,i) \in E} x_{(j,i)}^{md} = \begin{cases} R_m, & \text{for } i = s \\ -R_m, & \text{for } i = D \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

$$x_{(i,j)}^{md} \leq f_{(i,j)}^m \quad \forall d \in D \quad (2)$$

where (1) reflects the information flow balance equation similar to the physical flow balance equation. Equation (2)

specifies the network coding condition, relating physical rates to information rates.

Here, the physical flow rate vector  $\mathbf{f}$  is called the coding subgraph and can vary within a constraint set  $\mathbf{F}$ . For a feasible coding subgraph, [22, Theorem 1] states that the multicast sessions can be achieved with the distributed random network coding schemes in [7] and [17] using intra-session network coding. For a completeness of this paper, we include this theorem here with slight adaptation to our scenario:

*Theorem 1:* Given a feasible coding subgraph  $\mathbf{f} \in \mathbf{F}$ , a multicast session  $m$  at rate arbitrarily close to  $R_m$  from source node  $s$  to destination nodes in the set  $D$  and that injects packets at rate arbitrarily close to  $f_{(i,j)}^m$  on each link  $(i, j)$  is achievable with network coding if and only if the information flow rate vector  $\mathbf{x}$  and the physical flow rate vector  $\mathbf{f}$  satisfy (1) and (2).

Therefore, when setting up optimal multicast sessions over the hybrid network, there is no loss of optimality in separating the problems of subgraph selection and network coding. In other words, we can find an optimal coding subgraph  $\mathbf{f}$  satisfying (1) and (2), and then apply a network coding scheme to it where coding is done on overlapping links across different destination nodes' paths.

For each multicast session, we find multiple paths from the source node to destination nodes using existing multi-path routing schemes [30], [31]. For each node  $d \in D$ , we use a matrix  $H_d = \{h_{dj}^l\}$  to represent the relationship between its transmission paths and corresponding links. More specifically, suppose destination node  $d$  has  $J(d)$  alternative paths from source node  $s$ , then  $h_{dj}^l = 1$  if the path  $j$  of node  $d$  uses link  $l$ , and  $h_{dj}^l = 0$  otherwise.

When multi-path routing is employed, a link  $l$  might be shared by multiple paths of a certain destination node. Therefore, destination  $d$ 's bandwidth consumption for layer  $m$  on link  $l$  is the sum of the specified layer- $m$  flow rates on  $d$ 's all paths which pass link  $l$ . Mathematically, let  $R_{dj}^m$  denote the information flow rate of destination node  $d$ 's  $j$ th path in multicast session  $m$ , and  $f_l^m$  represent the physical flow rate for link  $l$  in multicast session  $m$ , then we have  $x_l^{md} = \sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m$ . The information flow balance condition in (1) will be automatically satisfied. With intra-session coding, the network coding constraint in (2) becomes

$$\sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m \leq f_l^m \quad \forall m \in M \quad \forall l \in E \quad \forall d \in D. \quad (3)$$

### C. Channel Capacity Model

In a wired network, the total transmission rate of the physical flow at each link should be no more than its capacity  $C_{(i,j)}$ , that is

$$\sum_{m \in M} f_{(i,j)}^m \leq C_{(i,j)} \quad \forall (i, j) \in E_1. \quad (4)$$

In wireless networks, however, the capacity of a wireless link is interrelated with other adjacent wireless links. Consequently, we should consider the wireless link contention in a shared transmission medium by introducing constraints of the location-dependent contention among the competing

wireless data flows [19]. In the proposed problem formulation, the assumption is that the wireless medium capacity  $C$  is shared among a wireless link  $l$  and the cluster of its competing links. The method in [20] considers the spatial locations of the nodes and determines which transmission can be successfully received by its intended recipient. According to this protocol model, suppose that any link originating from node  $k$  will interfere with link  $(i, j)$  if the link distance  $d_{(k,j)} < (1 + \Delta)d_{(i,j)}$ ,  $\Delta \geq 0$  and define  $\Psi_{(i,j)}$  for each link  $(i, j) \in E_2$  as the cluster of links that cannot transmit when link  $(i, j)$  is active. As compared to individual links in traditional wired network, the notation of cluster can be treated as a basic resource unit. Wireless data flows compete for the capacity of individual cluster that is equivalent to the capacity of the wireless shared medium. Hence, the wireless network channel capacity constraint [35] is

$$\sum_{m \in M} f_{(i,j)}^m + \sum_{(p,q) \in \Psi_{(i,j)}} \sum_{m \in M} f_{(p,q)}^m \leq C \cdot (1 - \rho_{(i,j)}) \quad \forall (i, j) \in E_2 \quad (5)$$

where  $C$  is defined as the maximum rate of link  $(i, j)$  and its corresponding cluster  $\Psi_{(i,j)}$  supported by the wireless shared medium,  $\rho_{(i,j)}$  is assumed to be the packet loss probability at wireless link  $(i, j)$ . Theoretically this packet loss rate can be derived from the Gilbert–Elliott model [41], [42].

### D. Rate-Distortion Model

From the perspective of application-layer QoS, rate-distortion related model [21] could be picked as the optimized targeted utility for video applications

$$D_e(R_e) = \frac{\theta}{R_e - R_0} + D_0 \quad (6)$$

where  $D_e$  is the distortion of the encoded video sequence, measured by the mean squared error, and  $R_e$  is the encoded rate. The variables  $\theta$ ,  $R_0$ , and  $D_0$  are the parameters of the R-D model, which can be fitted to empirical data from trial encodings using nonlinear regression techniques.

For an SVC stream, a destination node can subscribe to a partial layer. To characterize the video streaming performance of each layer  $m$ , we introduce a utility function  $U_m(\cdot)$ , which is continuously differentiable, increasing and strictly concave with respect to the receiving rate. In this paper, we multicast the video streams to all destination nodes and attempt to maximize the total utility of all recipients. In other words, our objective function is given by

$$\max \sum_{d \in D} \sum_{m \in M} U_m(R_d^m) = \max \sum_{d \in D} \sum_{m \in M} U_m \left( \sum_{j=1}^{J(d)} R_{dj}^m \right) \quad (7)$$

where  $R_d^m$  denotes the received rate at destination node  $d$  in multicast session  $m$ . Using the R-D model in (6) and Taylor expansion, we can define the utility function in (7) as the absolute value of the distortion decrement for destination node

$d$  when a new layer  $m$  is successfully received and decoded

$$U_m(R_d^m) = -[D_e(\sum_{i=0}^{m-1} R_d^i + R_d^m) - D_e(\sum_{i=0}^{m-1} R_d^i)] \quad (8)$$

$$\approx \frac{\theta}{(\sum_{i=0}^{m-1} R_d^i - R_0)^2} R_d^m - \frac{\theta}{(\sum_{i=0}^{m-1} R_d^i - R_0)^3} (R_d^m)^2.$$

Equation (8) is a quadratic function with regard to  $R_d^m$  and the coefficient of the quadratic term is a negative number, it can be easily seen that  $U_m(R_d^m)$  is strictly concave.

#### IV. PROBLEM FORMULATION

Within the context of SVC, layered multi-rate multicasting is equivalent to a generalized multi-source problem where the inter-layer dependency is considered. The proposed optimization problem will integrate the prior context of source decomposition into the layered multi-rate multicast optimization.

In most previous rate control schemes [9], [11], one sink node can either receive an entire layer or discard the layer, i.e., receiving a partial layer when there is residual bandwidth is not supported. In contrast, the proposed optimization formulation is a continuous optimization problem and makes a receiver possible to subscribe to a partial quality layer by introducing a tolerable rate region  $[b_m, B_m]$  for Layer  $m$ . Therefore, the proposed algorithm can fully utilize the network bandwidth resource. As follows, we propose a new optimization formulation which can achieve the global optimal solution while the layer dependency constraints of SVC stream are strictly satisfied with minimum costs

$$\mathbf{P1} : \max_{\mathbf{R}} \sum_{d \in D} \sum_{m \in M} U_m(\sum_{j=1}^{J(d)} R_{dj}^m) \quad (9)$$

s.t.

- 1)  $\sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m \leq f_l^m \quad \forall m \in M \quad \forall l \in E \quad \forall d \in D$
- 2)  $\sum_{m \in M} f_l^m \leq C_l \quad \forall l \in E_1$
- 3)  $\sum_{m \in M} f_l^m + \sum_{k \in \Psi(l)} \sum_{m \in M} f_k^m \leq C \cdot (1 - \rho_l) \quad \forall l \in E_2$
- 4)  $b_m \leq \sum_{j=1}^{J(d)} R_{dj}^m \leq B_m \quad \text{or} \quad \sum_{j=1}^{J(d)} R_{dj}^m = 0 \quad \forall m \in M \quad \forall d \in D$
- 5)  $\frac{\sum_{j=1}^{J(d)} R_{dj}^m}{b_m} \geq \frac{\sum_{j=1}^{J(d)} R_{dj}^{(m+1)}}{B_{(m+1)}} \quad \forall m \in \{1, 2, \dots, M-1\} \quad \forall d \in D$
- 6)  $R_{dj}^m \geq 0 \quad \forall j \in J(d) \quad \forall m \in M \quad \forall d \in D$
- 7)  $f_l^m \geq 0 \quad \forall l \in E \quad \forall m \in M.$

Constraint 1 specifies the required physical flow rate on each link for each layer under the network coding condition. With network coding, different destinations will not compete for link bandwidth within the same layer, therefore the required

physical flow rate on link  $l$  for Layer  $m$  is the largest information flow rate on link  $l$  consumed among all destination nodes. Constraint 2 ensures that for each wired link, the aggregate physical flow rates of different layers do not exceed the link's capacity. Constraint 3 characterizes the wireless link contention in a shared medium. For each wireless link  $l$ , the sum of  $l$ 's physical flow rate and the physical flow rates of links in  $\Psi(l)$  cannot exceed the wireless medium capacity  $C$ . Constraint 4 gives the lower bound and upper bound of the transmission rates allocated for Layer  $m$ , denoted by  $b_m$  and  $B_m$ , respectively. According to Proposition 1, constraints 4 and 5 together can strictly ensure that each destination node subscribes to SVC multicast layers in an incremental order, i.e., Layer  $m$  being received before Layer  $m+1$ . Constraint 6 specifies that the allocated rates are nonnegative.

*Proposition 1:* The allocated rate  $\mathbf{R}$  satisfies constraints 4 and 5 if and only if each destination node receives layered video streams in an incremental order.

*Proof:* See Appendix A. ■

To ensure the convexity of the proposed optimization problem **P1**, constraint 4 needs to be re-defined to meet the convexity requirement. Based on the nonnegativity constraint 6, it can directly imply  $\sum_{j=1}^{J(d)} R_{dj}^m \geq 0$  from  $R_{dj}^m \geq 0$ . Hereby, we can simply extend the second equality term  $\sum_{j=1}^{J(d)} R_{dj}^m = 0$  in constraint 4 to  $\sum_{j=1}^{J(d)} R_{dj}^m \leq 0$  because  $\sum_{j=1}^{J(d)} R_{dj}^m = 0$  can be promised along with the nonnegativity constraint 6. Therefore, constraint 4 is formalized as  $b_m \leq \sum_{j=1}^{J(d)} R_{dj}^m \leq B_m$ , or  $\sum_{j=1}^{J(d)} R_{dj}^m \leq 0$ , and further simplified as a cubic inequality  $(\sum_{j=1}^{J(d)} R_{dj}^m)(\sum_{j=1}^{J(d)} R_{dj}^m - b_m)(\sum_{j=1}^{J(d)} R_{dj}^m - B_m) \leq 0$ .

In **P1**, the optimization variable is the rate vector  $\mathbf{R} = [R_{dj}^m]$ , i.e., the information flow rates along the  $j$ th path to destination node  $d$  in multicast session  $m$ ,  $\forall m \in M, \forall d \in D, \forall j \in J(d)$ . If we choose for each destination node the same number of multiple paths, the total number of the optimization variables will be  $|M| \cdot |D| \cdot |J(d)|$ . On the other hand, considering optimizing over the entire network without specified paths from the source node to destination nodes, the optimization variable will be the links' information flow vector  $\mathbf{x} = [x_l^{m,d}]$ , i.e., the vector of information flow rate on link  $l$  for destination  $d$  of multicast session  $m$ , where  $\forall m \in M, \forall d \in D, \forall l \in E$  and accordingly, the total number of optimization variables is  $|M| \cdot |D| \cdot |E|$ . In large-scale networks, the number of total links  $|E|$  is often much larger than the number of transmission paths  $|J(d)|$ . Therefore, compared with performance optimization over the entire network, **P1** has a much smaller number of optimization variables.

We can see that **P1** has a unique optimal solution since its objective function is strictly concave and the solution space defined by the constraints is convex. In other words, this is a convex optimization problem. Centralized solutions require global information and coordination between all nodes and links, which is very costly and sometimes infeasible in practice [23], [24]. In the following section, we will develop a distributed solution based on decomposition and duality theories.

## V. DISTRIBUTED ALGORITHM

In this section, we develop a distributed solution to the proposed optimization problem which allows each node and link to control and update the transmission parameters by itself.

### A. Primal Decomposition

Decomposition theories provide a mathematical foundation for the design of modularized and distributed control of networks [24]. The decomposition procedure aims to decompose a large and complex optimization problem into a set of small sub-problems, which can be then solved with distributed and often iterative algorithms that converge to the global optimum. For **P1** with coupling variables  $f_l^m$ , the primal decomposition is often used

$$\begin{aligned} \mathbf{P1-1} : \quad & \max_{\mathbf{R}} \sum_{d \in D} \sum_{m \in M} U_m \left( \sum_{j=1}^{J(d)} R_{dj}^m \right) & (10) \\ & \text{s.t. constraints 1, 4, 5, and 6} \end{aligned}$$

$$\begin{aligned} \mathbf{P1-2} : \quad & \max_{\mathbf{f}} U^*(\mathbf{f}) & (11) \\ & \text{s.t. constraints 2, 3, and 7} \end{aligned}$$

where **P1-1** performs a low-level optimization when the coupling variable vector  $\mathbf{f}$  is fixed, while **P1-2** performs a high-level optimization to update  $\mathbf{f}$ .  $U^*(\mathbf{f})$  is the value of the objective function in **P1-1** for a given  $\mathbf{f}$ . The output of the low-level optimization is locally optimal and provides an approximation to the global optimal solution.

As mentioned in Section III-B, the coupling variable vector  $\mathbf{f}$  in **P1** represents the sub-graph of network coding. Note that in **P1**, the impact of network coding is embedded in constraint 1. Based on Theorem 1, it implies some form of ‘‘separation principle’’ that allows independent decisions on resource utilization and rate control during network coding. This suggests that the optimal configurations of multicast sessions over the hybrid network can be determined by decoupling the problem of subgraph selection from network coding. The task of the high-level optimization problem **P1-2** is to update  $\mathbf{f}$ , by selecting the optimal subgraph, while the low-level optimization problem **P1-1** attempts to find a locally optimal solution for resource utilization and rate control of a specified network coding scheme for a given coding subgraph  $\mathbf{f}$ .

### B. Low-Level Optimization

We observe that the low-level optimization problem **P1-1** can be further decoupled using dual decomposition. More specifically, by relaxing the coupling constraints 1, 4, and 5 with Lagrange multipliers  $\lambda$ ,  $\mu$ , and  $\eta$ , respectively, **P1-1** can be written as follows:

$$\begin{aligned} L(\mathbf{R}, \lambda, \mu, \eta) = & \sum_{d \in D} \sum_{m \in M} U_m \left( \sum_{j=1}^{J(d)} R_{dj}^m \right) \\ & - \sum_{l \in E} \sum_{d \in D} \sum_{m \in M} \lambda_d^{ml} \left[ \sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m - f_l^m \right] - \sum_{d \in D} \sum_{m \in M} \end{aligned}$$

$$\begin{aligned} & \mu_d^m \left[ \left( \sum_{j=1}^{J(d)} R_{dj}^m \right) \left( \sum_{j=1}^{J(d)} R_{dj}^m - b_m \right) \left( \sum_{j=1}^{J(d)} R_{dj}^m - B_m \right) \right] \\ & - \sum_{d \in D} \sum_{m=1}^{M-1} \eta_d^m \left[ \frac{\sum_{j=1}^{J(d)} R_{dj}^{(m+1)}}{B_{(m+1)}} - \frac{\sum_{j=1}^{J(d)} R_{dj}^m}{b_m} \right] \end{aligned} \quad (12)$$

and the corresponding Lagrange dual function is

$$g(\lambda, \mu, \eta) = \sup_{\mathbf{R}} L(\mathbf{R}, \lambda, \mu, \eta) \quad R_{dj}^m \geq 0. \quad (13)$$

The Lagrange dual problem of **P1-1** can be formulated as follows:

$$\min_{\lambda \geq 0, \mu \geq 0, \eta \geq 0} g(\lambda, \mu, \eta). \quad (14)$$

According to convex optimization theories [24], [25], if the original problem **P1-1** is convex, it is equivalent to its Lagrange dual problem in (14). Then, the low-level optimization problem **P1-1** can be further decomposed to a secondary master dual problem **P1-1a** and a set of sub-problems **P1-1b** that can be solved in a distributed manner

$$\begin{aligned} \mathbf{P1-1a} : \quad & \min_{\lambda, \mu, \eta} g(\lambda, \mu, \eta) & (15) \\ & \text{s.t. } \lambda \geq 0 \quad \mu \geq 0 \quad \eta \geq 0 \end{aligned}$$

$$\begin{aligned} \mathbf{P1-1b} : \quad & \max_{\mathbf{R}} L(\mathbf{R}, \lambda, \mu, \eta) & (16) \\ & \text{s.t. } R_{dj}^m \geq 0 \quad \forall j \in J(d) \quad \forall m \in M \quad \forall d \in D. \end{aligned}$$

At the lower level, the sub-problems **P1-1b** (i.e., the Lagrangians) for each  $d$ ,  $j$  and  $m$ , can be solved separately. At the higher level, we have the secondary master dual problem **P1-1a** to update dual variables  $\lambda$ ,  $\mu$ , and  $\eta$ .

Since the objective functions of problem **P1-1a** and **P1-1b** are differentiable with respect to the dual variables  $\lambda$ ,  $\mu$ ,  $\eta$  and primal variables  $\mathbf{R}$ , both problems can be solved by the gradient algorithm [26], [27]. Based on this observation, we propose the following primal-dual algorithm that updates the primal and dual variables simultaneously to solve the low-level optimization problem **P1-1**:

$$R_{dj}^m(t_L + 1) = [R_{dj}^m(t_L) + a(t_L) \frac{\partial L(\mathbf{R}, \lambda, \mu, \eta)}{\partial R_{dj}^m}]^+ \quad (17)$$

$$\lambda_d^{ml}(t_L + 1) = [\lambda_d^{ml}(t_L) - b(t_L) \frac{\partial L(\mathbf{R}, \lambda, \mu, \eta)}{\partial \lambda_d^{ml}}]^+ \quad (18)$$

$$\mu_d^m(t_L + 1) = [\mu_d^m(t_L) - c(t_L) \frac{\partial L(\mathbf{R}, \lambda, \mu, \eta)}{\partial \mu_d^m}]^+ \quad (19)$$

$$\eta_d^m(t_L + 1) = [\eta_d^m(t_L) - d(t_L) \frac{\partial L(\mathbf{R}, \lambda, \mu, \eta)}{\partial \eta_d^m}]^+ \quad (20)$$

where  $t_L$  denotes the iteration index,  $a(t)$ ,  $b(t)$ ,  $c(t)$  and  $d(t)$  are positive step sizes, and  $[\cdot]^+$  denotes the projection onto the set of non-negative real numbers.

In terms of their physical meanings,  $\lambda$  represents the ‘‘congestion prices’’ of information flow at all links, i.e.,  $\lambda_d^{ml}$  can

be considered as the ‘‘congestion price’’ of information flow at link  $l$  for destination node  $d$ 's bandwidth requirement in Layer  $m$ . At each link  $l$ , if the total information flow bandwidth demand  $\sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m$  in Layer  $m$  exceeds the supply  $f_l^m$ , then the ‘‘congestion price’’  $\lambda_d^{ml}$  will increase. As a result, in problem **P1-1b**,  $R_{dj}^m$  will decrease in order to meet the link's bandwidth requirement of information flow,  $f_l^m$ , and vice versa. Similarly, the other two Lagrange multipliers,  $\mu$  and  $\eta$ , can be considered as the ‘‘SVC encoding prices’’ for each destination node in a multicast session. Furthermore, all updating operations are distributed and can be implemented at individual links and nodes using only local information.

### C. High-Level Optimization

The low-level optimization and corresponding primal-dual algorithm operate under the assumption that the value of  $\mathbf{f}$  is fixed. Next, we discuss how to update  $\mathbf{f}$  in order to solve the high-level optimization problem **P1-2**. Suppose  $\hat{\lambda}_d^{ml}$  is the optimal Lagrange price, i.e., optimal variable corresponding to the constraint  $\sum_{j=1}^{J(d)} h_{dj}^l R_{dj}^m \leq f_l^m$  in **P1-1**. First we define the Lagrangian of **P1-2** as follows:

$$\begin{aligned} L'(\mathbf{f}, \alpha, \beta) &= U^*(\mathbf{f}) - \sum_{l \in E_1} \alpha_l \left( \sum_{m \in M} f_l^m - C_l \right) \quad (21) \\ &\quad - \sum_{l \in E_2} \beta_l \left( \sum_{m \in M} f_l^m + \sum_{k \in \Psi(l)} \sum_{m \in M} f_k^m - C \cdot (1 - \rho_l) \right) \\ &= U^*(\mathbf{f}) - \sum_{l \in E_1} \alpha_l \left( \sum_{m \in M} f_l^m - C_l \right) - \sum_{l \in E_2} \beta_l \sum_{m \in M} f_l^m \\ &\quad - \sum_{l \in E_2} \sum_{m \in M} f_l^m \left( \sum_{k \in \Phi(l)} \beta_k \right) + \sum_{l \in E_2} \beta_l C \cdot (1 - \rho_l) \end{aligned}$$

where we introduce a new notation  $\Phi(l)$  to denote the cluster of links that are interfered by link  $l$ . Since  $\Psi(k)$  denotes the cluster of all links that cannot be transmitted when link  $k$  is active, we have  $k \in \Phi(l) \iff l \in \Psi(k)$ . And the corresponding Lagrange dual function is

$$g'(\alpha, \beta) = \sup_{\mathbf{f}} L'(\mathbf{f}, \alpha, \beta) \quad f_l^m \geq 0 \quad \forall l \in E \quad \forall m \in M. \quad (22)$$

Similar to the solution of low-level optimization problem **P1-1**, with dual decomposition, the following procedure is used to solve the high-level optimization problem **P1-2**:

$$f_l^m(t_H + 1) = [f_l^m(t_H) + a'(t_H) \frac{\partial L'(\mathbf{f}, \alpha, \beta)}{\partial f_l^m}]^+ \quad (23)$$

$$\alpha_l(t_H + 1) = [\alpha_l(t_H) - b'(t_H) \frac{\partial L'(\mathbf{f}, \alpha, \beta)}{\partial \alpha_l}]^+ \quad (24)$$

$$\beta_l(t_H + 1) = [\beta_l(t_H) - c'(t_H) \frac{\partial L'(\mathbf{f}, \alpha, \beta)}{\partial \beta_l}]^+ \quad (25)$$

where  $t_H$  denotes the iteration index,  $a'(t)$ ,  $b'(t)$  and  $c'(t)$  are positive step sizes.

In terms of their physical meanings,  $\alpha$  and  $\beta$  can be considered as the ‘‘aggregated congestion prices’’ of physical flows at wired and wireless links, respectively. At each wired (or

wireless) link  $l$ , if the total physical flow bandwidth demand  $\sum_{m \in M} f_l^m$  (or  $\sum_{m \in M} f_l^m + \sum_{k \in \Psi(l)} \sum_{m \in M} f_k^m$ ) exceeds the supply  $C_l$  (or  $C \cdot (1 - \rho_l)$ ), then the ‘‘aggregate congestion price’’  $\alpha_l$  (or  $\beta_l$ ) will increase. As a result,  $f_l^m$  in **P1-2** will decrease in order to meet the link's bandwidth supply of physical flow,  $C_l$  (or  $C \cdot (1 - \rho_l)$ ), and vice versa. The update of  $f_l^m$  can be performed individually by each link, only with knowledge of the congestion price  $\alpha$  or  $\beta$ ; while the update of  $\alpha$  and  $\beta$  only uses the local information of each link.

### D. Convergence Analysis

We analyze the convergence behavior of the proposed algorithm. Regard the primal-dual algorithm as a nonlinear autonomous system on which we can apply the following Lyapunov stability theorem [26], [29].

*Theorem 2 (Lyapunov's Theorem):* Consider an autonomous system with its equilibrium point at  $\hat{x} = 0$ , this equilibrium point is globally asymptotically stable if there exists a continuously differentiable Lyapunov function  $V(x)$ , such that:

- 1)  $V(x) > 0$ ,  $\forall x \neq 0$ ; and  $V(x) = 0$ , when  $x = 0$ ;
- 2)  $\dot{V}(x) \leq 0$ ,  $\forall x$ ; and  $\dot{V}(x) < 0$ ,  $\forall x \neq 0$ ;
- 3)  $V(x) \rightarrow \infty$ , when  $\|x\| \rightarrow \infty$ .

Note that the above theorem also holds if the equilibrium point is  $\hat{x} \neq 0$ , by considering a system with state vector  $x - \hat{x}$ .

*Proposition 2:* If  $(\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta})$  is an equilibrium point of the low-level primal-dual algorithm outlined in (17)–(20), then the equilibrium point is asymptotically stable, i.e., the low-level primal-dual algorithm can converge to its equilibrium point,  $(\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta})$ .

*Proof:* See Appendix B. ■

*Proposition 3:* If  $(\hat{\mathbf{f}}, \hat{\alpha}, \hat{\beta})$  is an equilibrium point of the high-level primal-dual algorithm proposed in (23)–(25), then the equilibrium point is asymptotically stable.

*Proof:* See Appendix C. ■

According to the Lyapunov's theorem, if we can find a Lyapunov function for the dynamical system which satisfies all of these three conditions, then the equilibrium point of the dynamic system is asymptotically stable. From Propositions 2 and 3, we proved the global asymptotic stability of the primal and dual controllers of (17)–(20) and (23)–(25), respectively, which leads to the convergence behavior of the distributed solution to the dual problems of **P1-1** and **P1-2**. Since **P1-1** and **P1-2** are both convex, we can solve them through their equivalent dual problems using the proposed distributed algorithms [24].

### E. Summary of the Distributed Algorithm and Its Implementation

To implement the proposed distributed algorithm, each link  $l$  or destination node  $d$  is treated as an entity capable of processing, storing, and communicating information. In practice, each link  $l = (i, j)$  is delegated to its sender node  $i$ , and all computations related to link  $l = (i, j)$  will be executed on node  $i$ . Assume that the processor for link  $l$  keeps track of variables  $f_l^m$ ,  $\alpha_l$ ,  $\beta_l$  and  $\lambda_d^{ml}$ , while the processor of destination node  $d$  keeps track of variables  $R_{dj}^m$ ,  $\mu_d^m$  and  $\eta_d^m$ . A distributed version of the proposed algorithm can be summarized in Algorithm 1.

Note that the low-level and high-level optimizations operate at different timescales. The low-level iteration algorithm is

**Algorithm 1** Distributed two-level optimization algorithm

**Step 1. Initialization:**

Set  $R_{d_j}^m(0)$ ,  $\lambda_d^m(0)$ ,  $\mu_d^m(0)$ ,  $\eta_d^m(0)$ ,  $f_l^m(0)$ ,  $\alpha_l(0)$  and  $\beta_l(0)$ , respectively, to some nonnegative value for all  $d, m, l$  and  $j$ .

**Step 2. Low-level optimization** ( $t_L = 1, 2, \dots$ ):

**At link  $l$ :**

- 1) Receives  $R_{d_j}^m(t_L)$  from the subset  $\{d|d \in D \text{ and } h_{d_j}^l = 1\}$ .
- 2) Fetches  $f_l^m(t_H)$  stored in the local processor of node  $i$ .
- 3) Updates the congestion price  $\lambda_d^m(t_L)$  according to (18).
- 4) Sends the control packet (CP) that comprises updated price  $\lambda_d^m(t_L + 1)$  in the downstream direction to the subset of destination nodes  $\{d|d \in D \text{ and } h_{d_j}^l = 1\}$ .

**At destination node  $d$ :**

- 1) Receives from the network the aggregate congestion price  $\sum_{l \in E} \lambda_d^m(t_L) \cdot h_{d_j}^l$ .
- 2) Fetches  $\mu_d^m(t_L)$  and  $\eta_d^m(t_L)$  stored in the local processor.
- 3) Updates allocated rate  $R_{d_j}^m(t_L)$  with (17).
- 4) Updates the SVC encoding price  $\mu_d^m(t_L)$  with (19).
- 5) Updates the SVC encoding price  $\eta_d^m(t_L)$  with (20).
- 6) Sends the rate packet (RP) that contains the updated rate  $R_{d_j}^m(t_L + 1)$  in the upstream direction to the subset of links  $\{l|l \in E \text{ and } h_{d_j}^l = 1\}$ .

Iterate until the low-level implementation converges to the optimality or the maximum iteration number is achieved, then proceed to Step 3 (the high-level optimization stage).

**Step 3. High-level optimization** ( $t_H = 1, 2, \dots$ ):

**At wired link  $l \in E_1$ :**

- 1) Receives from the network the locally optimal congestion price  $\hat{\lambda}_d^m$  of the low-level implementation.
- 2) Fetches  $f_l^m(t_H)$  and  $\alpha_l(t_H)$  stored in the local processor.
- 3) Updates a new  $f_l^m(t_H)$  with (23).
- 4) Updates the aggregate congestion price  $\alpha_l(t_H)$  with (24).

**At wireless link  $l \in E_2$ :**

- 1) Receives from the network the locally optimal congestion price  $\hat{\lambda}_d^m$  of the low-level implementation.
- 2) Receives  $f_k^m(t_H)$  from the cluster  $\{k|k \in \Psi(l)\}$ .
- 3) Receives  $\beta_k(t_H)$  from the cluster  $\{k|k \in \Phi(l)\}$ .
- 4) Fetches  $f_l^m(t_H)$  and  $\beta_l(t_H)$  stored in the local processor.
- 5) Updates a new  $f_l^m(t_H)$  with (23).
- 6) Updates the aggregate congestion price  $\beta_l(t_H)$  with (25).
- 7) Transmits the flow-rate packet (FP) that contains the updated physical flow rate  $f_l^m(t_H + 1)$  to the cluster  $\{k|k \in \Phi(l)\}$ .
- 8) Transmits control packet (CP) that comprises the updated price  $\beta_l(t_H + 1)$  to the cluster  $\{k|k \in \Psi(l)\}$ .

If the high-level implementation converges to the optimality or the maximum iteration number is achieved, the algorithm stops, else go back to Step 2 (the low-level implementation).

in the inner loop and operates at a smaller time index  $t_L$ , while the high-level iteration algorithm is in the outer loop and performs at a larger time index  $t_H$ . More specifically, the high-level algorithm will not move to its step until  $\lambda_d^m(t_L)$  at the low-level converges to its optimum value  $\hat{\lambda}_d^m$ .

In summary, the centralized approach requires all of the above variables shared in the entire network and thus causes a great amount of communication overhead. Utilizing the proposed distributed algorithm, however, the communication overhead only comprises the sending overhead of  $\lambda_d^m(t_L + 1)$  and  $R_{d_j}^m(t_L + 1)$  at each iteration of the low-level optimization and the transmitting overhead of  $f_l^m(t_H + 1)$  and  $\beta_l(t_H + 1)$  at each iteration of the high-level optimization.

The overhead of the proposed distributed algorithm consists of two parts: the network coding overhead and the communication overhead. It is demonstrated in [17] that the

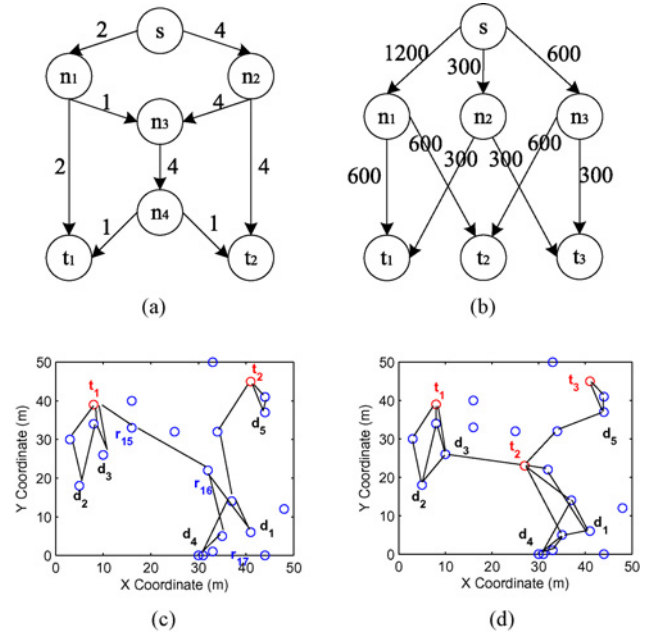


Fig. 3. Network topology associated with link capacity, where (a) and (c) illustrate the wired and wireless network in numerical experiment, respectively, (b) and (d) illustrate the wired and wireless network in packet level simulation, respectively.

side information required by network coding is very small, e.g., approximately 3% in the typical Internet scenario. The communication overhead are the CP and RP information in the low-level optimization, and the FP and CP information in the high-level optimization. Consider the implementation issues [39] and take Fig. 3(c) for example, at the end of each low-level iteration, link  $(r_{15}, r_{16})$  needs only to send its CP downward to destination nodes  $d_1$  and  $d_4$ , and destination node  $d_1$  needs only to send its RP upward to the links belonging to its six paths. Supposing each updated price or rate is float type that takes up 4 byte, and let  $M = 3$ , then the CP of link  $(r_{15}, r_{16})$  requires  $3 \times 2 \times 4 = 24$  bytes and the RP of node  $d_1$  requires  $3 \times 6 \times 4 = 72$  bytes. Generally, in the worst case, the CP of link  $l$  takes up  $4 \cdot M \cdot |D|$  bytes, and the RP of destination node  $d$  takes up  $4 \cdot M \cdot |J(d)|$  bytes. Similarly in the high-level optimization, wireless link  $l$  needs to send its FP and CP to the cluster  $\{k|k \in \Phi(l)\}$  and  $\{k|k \in \Psi(l)\}$ , respectively. Therefore, FP and CP require  $3 \times 4 = 12$  bytes and  $1 \times 4 = 4$  bytes, respectively. Thus, the total communication overhead sums up to 112 bytes. For the internet configuration of IP packet with a packet size of 1400 bytes, the communication overhead introduced by the proposed algorithm is  $112/1400 = 8\%$ . Furthermore, it can be noted that these packets (CPs, RPs and FPs) in practical implementation need not be communicated as separate packets; the CPs can be conveyed through a field in the video data packets, while the RPs and FPs can be conveyed through a field in the acknowledgement packets. The maximal additional delay introduced by sending these packets is the one way propagation delay of the particular multicast destination node.

The proposed distributed algorithm needs to be implemented whenever the initial network starts multicast session or the dynamic change of network condition suddenly happens, to catch up with the optimal allocated transmission rates for



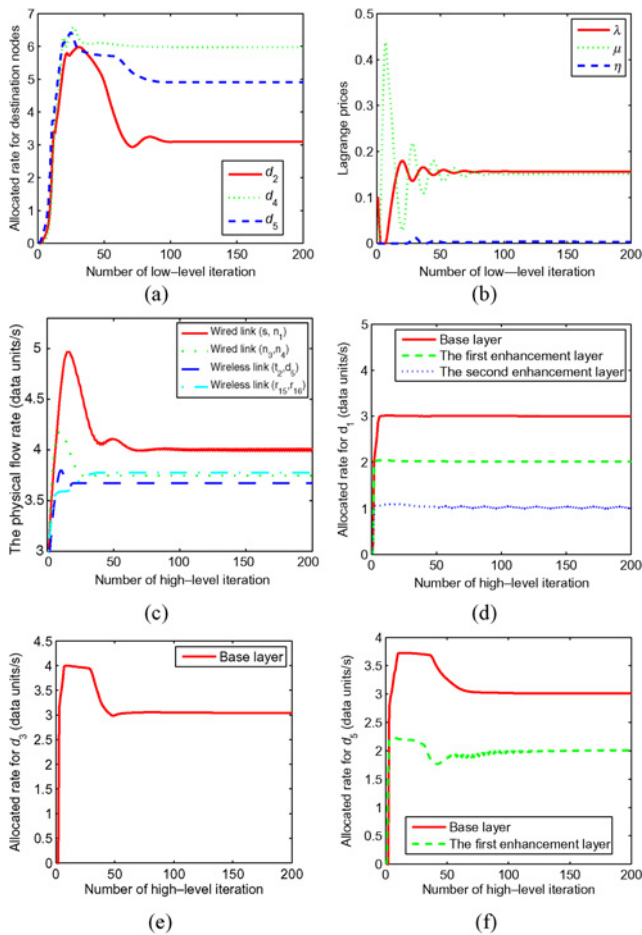


Fig. 4. Convergence performance of low-level optimization: (a) allocated rate for  $d_2$ ,  $d_4$ , and  $d_5$ , (b) example of Lagrange prices  $\lambda$ ,  $\mu$ , and  $\eta$ . And the convergence behavior of high-level optimization: (c) physical flow rate for links, (d) allocated rate for  $d_1$ , (e) allocated rate for  $d_3$ , and (f) allocated rate for  $d_5$ .

destination nodes. The implementation would not stop until the algorithm converges to the optimality or the maximum high-level iteration number is achieved. Theoretically, the convergence time spent by destination nodes to catch up with the calculated optimal transmission rates is the number of iterations multiplies the update interval of each iteration. It is found by [37] that an update interval which is about two to three times the one way propagation delay of the particular multicast destination node is sufficient.

## VI. EXPERIMENTAL RESULTS

In this section, experimental results are presented to demonstrate the performance of the proposed algorithm. We first conduct the numerical experiment in C++ code to jointly implement the iterations of both low-level and high-level optimization. The two-level optimization model is numerically solved to evaluate the convergence behavior of the proposed distributed algorithm and demonstrate that it is able to achieve the max-flow throughput. Packet level simulations are performed on ns-2 [32] with a hybrid wired and wireless network where SVC video streams are distributed. We show that the

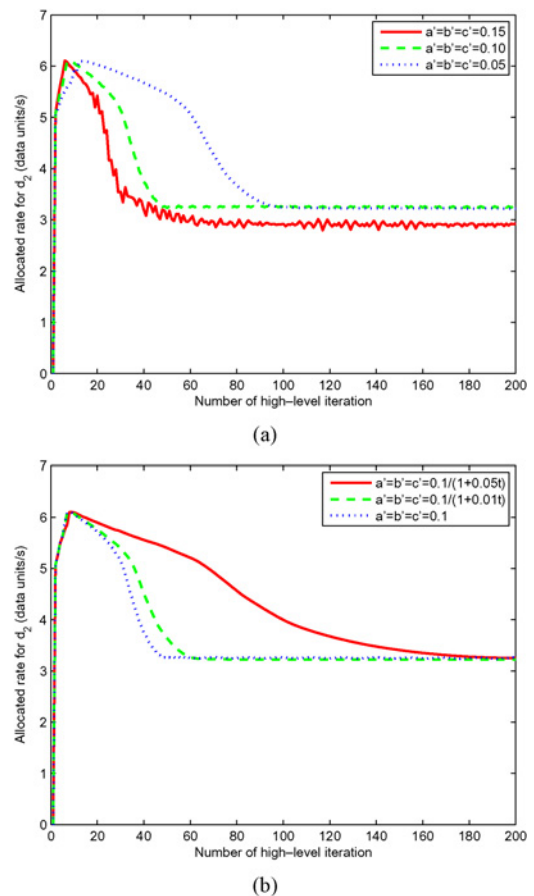


Fig. 5. Impact of step size on the convergence behavior. (a) Constant step sizes. (b) Diminishing step sizes and constant step size.

proposed algorithm will significantly improve the overall video quality. Furthermore, we study the impact of playback deadline (PD) and background traffic on the overall performance.

### A. Algorithm Behavior and Performance Evaluations

We conduct numerical experiments and evaluate the proposed algorithm over a hybrid wired and wireless network shown in Fig. 3. Fig. 3(a) shows the wired network by a typical butterfly network with network coding. Here,  $s$  denotes the source node,  $n_1, n_2, \dots, n_4$  denote relay nodes,  $t_1$  and  $t_2$  denote two receiver nodes in the wired network as well as the source nodes in the wireless network. Fig. 3(c) shows the wireless network, which consists of 20 randomly distributed wireless sensors.  $t_1$  and  $t_2$  denote the relay stations that connect the wireless network with the wired network,  $d_1, d_2, \dots, d_5$  denote 5 destination nodes and other nodes are wireless relay nodes. In Fig. 3(a), we can see that each relay station in the wired network has three alternative paths from the source. In Fig. 3(c), we assume that every two nodes within a distance of less than 20 m are able to communicate with each other. We plot two shortest paths for each destination node from relay stations. Therefore, each destination node has six alternative paths from the source.

In the numerical experiments, we assume that the video bit stream has three layers, with the base layer at a rate of 3 (data units/s), the first enhancement layer at a rate of 2 (data units/s)

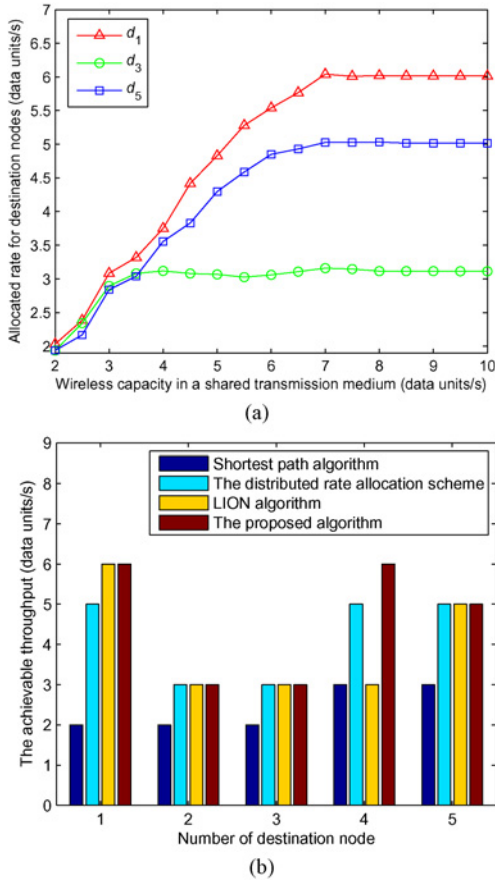


Fig. 6. (a) Allocated rate when wireless link contention exists. (b) Comparison of the achievable throughput by different algorithms.

and the second enhancement layer at a rate of 1 (data units/s). Suppose that the capacity of wireless links' cluster is  $C = 10$  with packet loss probability  $\rho_l = 0.1$ . Using the proposed algorithm, we can obtain that the achievable throughput for destination nodes  $d_1, d_2 \dots d_5$  are 6, 3, 3, 6, and 5, respectively. Therefore, destination node  $d_1$  and  $d_4$  can receive all the three SVC video stream layers,  $d_5$  can receive the lower two layers, while  $d_2$  and  $d_3$  can only receive the base layer.

Fig. 4 reflects the convergence behavior of the low-level optimization with a fixed step size  $a(t_L) = b(t_L) = c(t_L) = d(t_L) = 0.05$ , where Fig. 4(a) shows the allocated data rate for  $d_2, d_4$ , and  $d_5$ , and Fig. 4(b) shows the values of Lagrange prices  $\lambda, \mu$ , and  $\eta$ . It can be seen that all allocated data rates and Lagrange prices approach the optimal value after 90 iterations. This implies that the low-level optimization algorithm converges quickly. For example, the allocated data rate for  $d_4$  approaches within 10% of its optimal value after 30 iterations and converges to its optimal value 6.014 after 90 iterations. The Lagrange price  $\lambda$  approaches within 10% of its optimal value after 50 iterations and converges to its optimal value 0.151 after 90 iterations.

Fig. 4(c)–(f) shows the convergence behavior of the high-level optimization process. Fig. 4(d)–(f) shows the allocated data rate for  $d_1, d_3$ , and  $d_5$ , respectively, and Fig. 4(c) shows the physical flow for selected wired and wireless links. It can be seen that all allocated data rates and flow rates on links

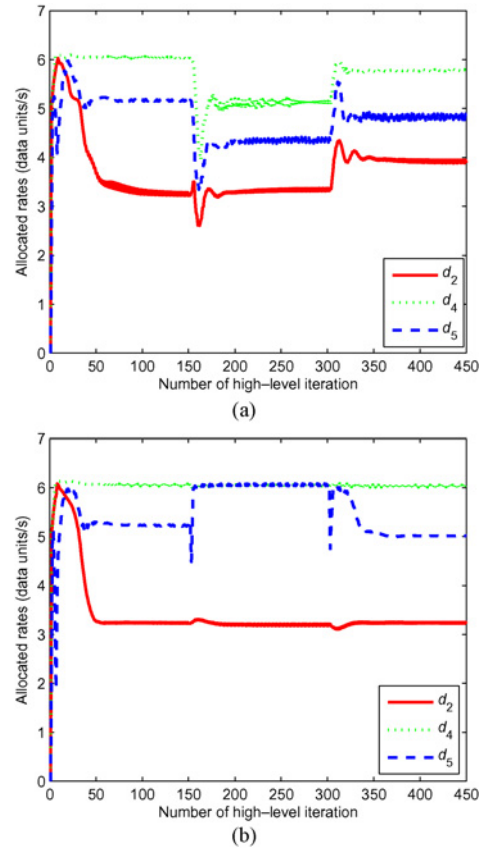


Fig. 7. High-level convergence behavior of the practical algorithm execution in dynamic environments. (a) Dynamic change of the maximum rate of wireless link  $C$ . (b) Wireless node  $r_{16}$  joins and leaves.

converge after 70 iterations. As discussed in Section V-E, the low-level optimization operates at a smaller timescale and needs a fast convergence speed. But the high-level counterpart is at the outer loop of the iteration, which operates at a larger timescale. So its convergence performance is determined by the inner loop of the low-level optimization. Therefore, we can indirectly control the convergence speed of the outer loop by adjusting that of the inner loop.

Next, we study the impact of step size on the convergence behavior of destination nodes' rate. Fig. 5(a) shows the convergence behavior of  $d_2$ 's allocated rate with three different step sizes, 0.15, 0.10, and 0.05. It can be seen that the convergence behavior is smoother but slower when the step size is smaller. With more detail, Fig. 5(b) shows the convergence behavior of  $d_2$ 's allocated rate with a fixed step size 0.1 and two diminishing step sizes  $0.1/(1 + 0.05t_H)$  and  $0.1/(1 + 0.01t_H)$ , both of which satisfy the requirement that  $\lim_{t_H \rightarrow +\infty} a'(t_H) = 0$  and  $\sum_{t_H=0}^{+\infty} a'(t_H) = +\infty$ . It can be seen that the convergence behavior with a diminishing step size  $0.1/(1 + 0.05t_H)$  is smoother but converges more slowly than a fixed step size. A major advantage using the varying step size is to be able to converge quickly first with large step sizes and then refine itself in the later stage with smaller step sizes. In conclusion, although a fixed step size is more convenient in practice and can converge more quickly, a varying step size is recommended since a slow-change rate profile is critical

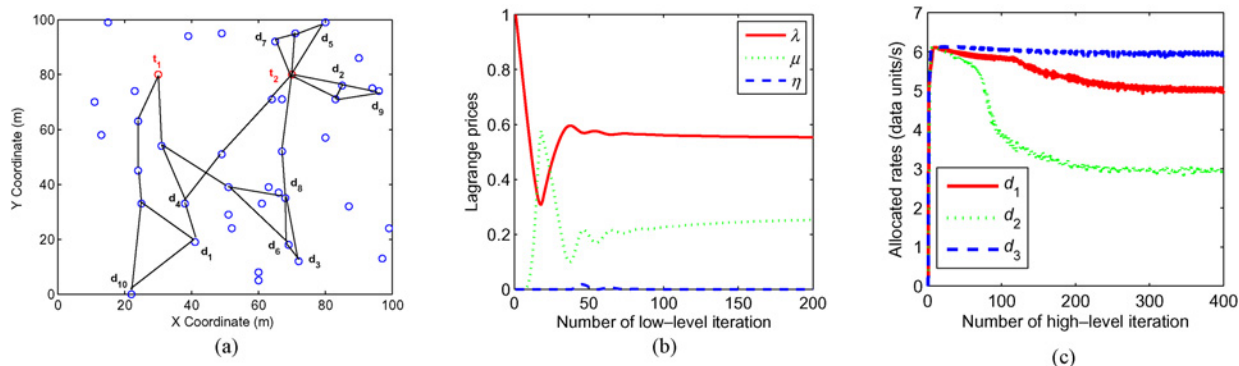


Fig. 8. (a) Larger wireless network with 10 destination nodes out of 40 randomly distributed wireless nodes. (b) Convergence behavior of low-level optimization. (c) Convergence behavior of high-level optimization.

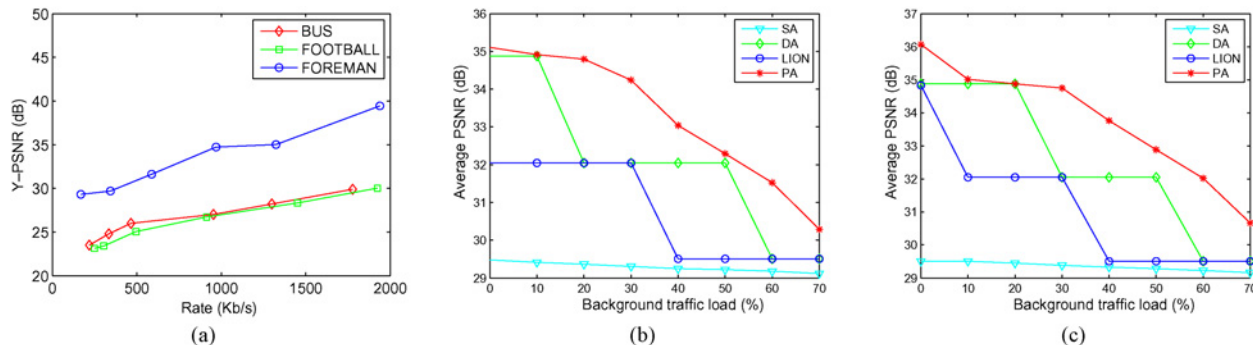


Fig. 9. (a) PSNR performance achieved by SVC stream for three CIF sequences with frame rate of 30 frames/s and GOP length of 32. And received video quality in PSNR of *FOREMAN* sequence for destination node  $d_5$  when the background traffic load varies from 0 to 70%, where the palyout deadline is set to (b) 500 ms and (c) 700 ms.

for video quality smoothness. Otherwise, a sudden change of access data rate will often result in undesirable quality fluctuation.

To study the condition when wireless link contention occurs, we vary the wireless capacity in a shared link cluster  $C$  from 10 to 2 and see the corresponding impact on the allocated rate. The results are shown in Fig. 6(a). As mentioned before, the achievable throughput for  $d_1$ ,  $d_3$ , and  $d_5$  with no link contention are 6, 3, and 5, respectively. When the wireless capacity  $C$  is less than 7, wireless link contention will first affect the rate for  $d_1$ . As  $C$  decreases, the allocated rate for  $d_5$  will decrease accordingly when the wireless capacity  $C$  is less than 6. And the allocated rate for  $d_3$  will be affected when  $C$  is less than 3.5. For example, if  $C$  is set to be 4, which means the link contention exists in the wireless network and it will influence  $d_1$  and  $d_5$ . Using the proposed algorithm, we can obtain that the achievable throughput for each destination nodes  $d_1$ ,  $d_3$ , and  $d_5$  can approach to 3.6, 3, and 3.5, respectively.

Fig. 6(b) shows the achievable throughput of all destination nodes  $d_1, d_2, \dots, d_5$ , obtained by four different methods: the shortest path algorithm, the distributed rate allocation scheme in [9], the LION algorithm [11], and the proposed algorithm. Compared with the other three multi-path algorithms, the achievable throughput of shortest path algorithm is usually smaller since each destination node only has one single path from the source node, which under-utilizes the available

network resource. Furthermore, it can be seen that network coding offers a significant gain in throughput. For example, at most time, the throughput achieved by the LION algorithm and the proposed algorithm (with network coding) is greater than that by the distributed rate allocation scheme (without network coding). Overall, it can be seen that the proposed algorithm that combines network coding technique and appropriate multi-path rate allocation mechanism can approximate the max-flow capacity for each member of a multicast group.

To investigate the convergence in dynamic environment [43], the execution of the proposed distributed algorithm always starts directly from Step 2 (low-level optimization step) with the previous optimal values (rather than from the initialization step when the values of all optimization variables and Lagrange variables are initialized to the very beginning), thus expediting its convergence. In Fig. 7(a), we study the convergence behavior of the running under a dynamic change of  $C$  (the maximum rate of wireless link). Initially, the optimization iteration begins from the initialization step with  $C = 6$ , and allocated rates for  $d_2$ ,  $d_4$  and  $d_5$  converge to the optimal values after approximate 70 iterations. At iteration 150 and 300,  $C$  is varied to 4 and 5, respectively, and the proposed algorithm starts directly from Step 2 with the previous optimal values. The number of additional iterations required to converge to new optimal values from the previous ones are about 25 and 35, respectively. The adaptation of the practical algorithm

TABLE I  
RECEIVED AVERAGE VIDEO QUALITY IN PSNR OF *BUS*, *FOOTBALL*, AND *FOREMAN* SEQUENCES WITH DIFFERENT PD

PSNR		<i>BUS</i> Sequence				<i>FOOTBALL</i> Sequence				<i>FOREMAN</i> Sequence			
PD (ms)		300	400	500	600	300	400	500	600	300	400	500	600
$d_1$	SA	23.2	23.9	23.9	23.9	22.9	23.2	23.2	23.2	29.4	29.5	29.5	29.5
	DA	23.9	26.4	27.7	27.7	23.2	25.6	27.4	27.4	29.5	32.0	34.9	34.9
	LION	23.6	23.9	26.4	26.4	23.0	23.2	25.6	25.6	29.4	29.5	32.0	32.0
	PA	23.6	26.3	26.9	27.4	23.0	25.6	26.6	27.2	29.4	32.0	34.0	34.8
$d_2$	SA	23.2	23.9	23.9	23.9	22.9	23.2	23.2	23.2	29.4	29.5	29.5	29.5
	DA	23.9	23.9	26.4	26.4	23.2	23.2	25.6	25.6	29.5	29.5	32.0	32.0
	LION	21.8	23.9	26.4	26.4	22.2	23.2	25.6	25.6	29.1	29.5	32.0	32.0
	PA	21.8	26.1	26.6	26.8	22.2	25.0	26.0	26.5	29.1	30.9	32.8	33.7
$d_3$	SA	23.9	23.9	23.9	23.9	23.2	23.2	23.2	23.2	29.5	29.5	29.5	29.5
	DA	23.9	26.4	27.7	27.7	23.2	25.6	27.4	27.4	29.5	32.0	34.9	34.9
	LION	26.4	27.7	30.4	30.6	25.6	27.4	30.0	30.2	32.0	34.9	39.2	39.7
	PA	27.2	29.4	30.4	30.6	27.0	29.0	30.0	30.2	34.7	37.3	39.2	39.7
$d_4$	SA	23.2	23.9	23.9	23.9	22.9	23.2	23.2	23.2	29.4	29.5	29.5	29.5
	DA	23.9	26.4	27.7	27.7	23.2	25.6	27.4	27.4	29.5	32.0	34.9	34.9
	LION	23.9	26.4	26.4	27.7	23.2	25.6	25.6	27.4	29.5	32.0	32.0	34.9
	PA	24.3	26.5	27.3	28.2	23.3	25.9	27.1	27.9	29.6	32.5	34.8	35.0
$d_5$	SA	22.7	23.4	23.8	23.9	22.6	22.9	23.1	23.2	29.3	29.4	29.5	29.5
	DA	23.9	26.4	27.7	27.7	23.2	25.6	27.4	27.4	29.5	32.0	34.9	34.9
	LION	23.9	26.4	26.4	27.7	23.2	25.6	25.6	27.4	29.5	32.0	32.0	34.9
	PA	26.0	27.2	28.3	28.8	24.9	27.0	28.0	28.3	30.7	34.8	35.1	35.9

execution to node joining and departure is shown in Fig. 7(b). The optimization iteration under network topology in Fig. 3(a) and (c) starts from the initialization step, and converges to the optimal values after approximate 70 iterations. At iteration 150 and 300, wireless node  $r_{16}$  joins and leaves the path of  $d_5$ , respectively, and the proposed algorithm starts directly from Step 2 with the previous optimal values. The number of additional iterations required to converge to new optimal values from the previous ones are about 20 and 30, respectively.

Compared with the algorithm running from the initialization step, Fig. 7 demonstrates that the dynamic execution of the proposed algorithm which starts from Step 2 with the previous optimal values converges much faster. In fact, it can also be observed that in all cases the convergence to 90% optimality is much faster, with at least 20% fewer iterations than those required for convergence to optimality. Therefore, we can obtain a feasible solution in realistic dynamic networks to a certain degree of optimality in a much shorter convergence time.

To evaluate the effect for a more general network, we implement the proposed algorithm over the hybrid network which consists of Figs. 3(a) and 8(a) with a more complex wireless network. In Fig. 8(a), we randomly place 40 wireless nodes in a 100 m-by-100 m square area and choose ten nodes to be destination nodes. Theoretically, since the proposed algorithm is implemented in a distributed manner that only requires local information, the communication overhead and thus convergence behavior would not be affected very much by larger scale of networks. For instance, Fig. 8(b) and (c) illustrates the convergence behavior of both low-level and high-level optimization. It can be seen that the convergence speed slows down slightly as the scale of networks increases. Therefore, the impact on the system performance affected by larger scale networks is relatively small.

## B. Packet Level Simulations

We use ns-2 to conduct packet level simulations on the network topology shown in Fig. 3(b) and (d). The wired network is composed of one source  $s$ , three relay nodes  $n_1, n_2, n_3$ , and three relay stations  $t_1, t_2, t_3$ . The wireless network has 20 randomly distributed sensor nodes within a 50 m-by-50 m area, among which 5 sensor nodes are considered as destinations. In the wired network, each relay station has two different paths from the source node. In the wireless network, we find for each destination node three different paths which are the three shortest paths from relay stations. In subsequence, each destination node has six alternative paths from the source node to obtain the SVC video stream.

We use Joint Scalable Video Model 7\_10 reference codec of H.264/AVC scalable extension standard [33], with three well-known test video sequences (*BUS*, *FOOTBALL*, and *FOREMAN*) at frame rate of 30 frames/s, CIF ( $352 \times 288$ ) resolution, and a GOP-length of 32 frames. They are encoded with 256 kb/s on the base layer, and 384 kb/s, 512 kb/s and 1024 kb/s on the enhancement layers by fine granularity scalable coding. The layered SVC sources are segmented into small blocks which are encapsulated to multiple packets with the packet size 512 bytes. The transmission delay for each link is set to be 50 ms. Fig. 9(a) shows the R-D performance (PSNR) of SVC for three CIF video sequences obtained from the packet-level simulation on ns-2. Using the proposed algorithm, we also obtained that the max-flow capacity for destination nodes  $d_1, d_2 \dots d_5$  are {1200, 900, 2100, 1200, 1800} kb/s. In the following, we will compare the simulation results obtained by the proposed algorithm against those by other algorithms for different application requirements.

We change the PD from 300 ms to 600 ms and evaluate the results obtained by the four different algorithms: shortest path algorithm (SA), distributed rate allocation algorithm (DA)

in [9], the LION algorithm in [11], and proposed algorithm (PA). The results are shown in Table I. Here, packets are considered to be lost if they do not arrive at the destination node before the scheduled playback time [36]. From Table I, we can see that the average packet delay of SA and DA (without network coding) is smaller than that of LION and PA. However, when the PD is set to large enough, e.g., 600 ms, the average video quality achieved by LION and PA is better than SA and DA due to the use of network coding. By allowing to receive partial layers, the proposed algorithm ensures the network's capacity resource is fully exploited, since if the residual bandwidth is not enough for an entire Layer  $m$  but still larger than  $b_m$ , Layer  $m$  can be partially received to improve the received video quality. On the contrary, when the same situation happens to the other three algorithms, Layer  $m$  will be discarded. Therefore, it can be seen that PA generally achieves the best overall video quality for all destination nodes.

To study the impact of background traffic on the resource allocation results with these four algorithms, *FOREMAN* is adopted as the test video sequence and the percentage of background traffic varies from 0 to 70% while the PDs are set to be 500 ms and 700 ms. The results are shown in Fig. 9(b) and (c). Since a layer is either entirely received or discarded by SA, DA, and LION algorithms, the received video quality achieved by these three algorithms can only locate in several discrete PSNR values. Furthermore, it can also be seen that with all the four algorithms and at different PDs, the received video quality in PSNR for  $d_5$  decreases as the increment of background traffic load. By allowing partial layer reception, PA outperforms the other three algorithms with better overall video quality for  $d_5$  at different background traffic loads and PDs.

## VII. CONCLUSION

In this paper, we developed an efficient flow control and resource allocation scheme for multicast of scalable video streaming over hybrid wired/wireless networks. In order to perform layered utility maximization over tiered wired/wireless coded networks, we proposed a joint source-network flow optimization scheme where each destination node accesses progressive layered stream in an incremental order. It aims to determine optimal multicast paths at associated rates for the highest sustainable layered video quality with minimum costs. A distributed algorithm with two levels of optimization has also been developed and illustrated in detail. Experimental results demonstrate that the proposed algorithm not only provides better video quality with optimal layered access for heterogeneous receivers, but benefits the asymptotic capacity of a hierarchical "hybrid" wired/wireless architecture in a scalable sense.

### APPENDIX A PROOF OF PROPOSITION 1

First, we prove that the sufficiency of constraints 4 and 5. From constraint 4, we can see that there only exist two situations for destination node  $d$  in multicast session  $m$ , i.e.,

$b_m \leq \sum_{j=1}^{J(d)} R_{dj}^m \leq B_m$  specifies that destination  $d$  subscribes to layer  $m$ , and  $\sum_{j=1}^{J(d)} R_{dj}^m = 0$  means that destination  $d$  fails to subscribe to layer  $m$ . Under constraint 5, we can have  $\forall m \in M, \forall d \in D$ , if  $b_m \leq \sum_{j=1}^{J(d)} R_{dj}^m \leq B_m$ , then  $\sum_{j=1}^{J(d)} R_{dj}^{(m+1)}$  can be either 0 or within the range of  $[b_{(m+1)}, B_{(m+1)}]$ ; if  $\sum_{j=1}^{J(d)} R_{dj}^m = 0$ , then  $\sum_{j=1}^{J(d)} R_{dj}^{(m+1)}$  can only be 0. It indicates that the destination  $d$  needs to join all the lower layers (from layer 1 to layer  $m$ ) before joining layer  $m+1$ , which guarantees that each destination node can receive layered video streams in an incremental order under constraints 4 and 5.

The necessity of constraints 4 and 5 can be easily proved in the following way. If each destination node receives layered video streams in an incremental order, then the lower bound and upper bound of the receiving rate required for each layer given by constraints 4 should be satisfied and the inequality in constraints 5 as well can be obviously obtained.

### APPENDIX B PROOF OF PROPOSITION 2

For the primal-dual method in low-level optimization, we construct the following Lyapunov function:

$$\begin{aligned} V(\mathbf{R}, \lambda, \mu, \eta) = & \sum_{d \in D} \sum_{m \in M} \left[ \sum_{j \in J(d)} \int_{\hat{R}_{dj}^m}^{R_{dj}^m} \frac{1}{a(\kappa)} (\kappa - \hat{R}_{dj}^m) d\kappa \right. \\ & + \sum_{l \in E} \int_{\hat{\lambda}_d^{ml}}^{\lambda_d^{ml}} \frac{1}{b(\vartheta)} (\vartheta - \hat{\lambda}_d^{ml}) d\vartheta + \int_{\hat{\mu}_d^m}^{\mu_d^m} \frac{1}{c(v)} (v - \hat{\mu}_d^m) dv \\ & \left. + \int_{\hat{\eta}_d^m}^{\eta_d^m} \frac{1}{d(\varsigma)} (\varsigma - \hat{\eta}_d^m) d\varsigma \right]. \end{aligned} \quad (26)$$

Note that  $V(\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta}) = 0$ . If  $R_{dj}^m \neq \hat{R}_{dj}^m$ , then  $\int_{\hat{R}_{dj}^m}^{R_{dj}^m} \frac{1}{a(\kappa)} (\kappa - \hat{R}_{dj}^m) d\kappa > 0$ , and this inequality can be extended to the other three arguments. Hence, we have  $V(\mathbf{R}, \lambda, \mu, \eta) > 0, \forall (\mathbf{R}, \lambda, \mu, \eta) \neq (\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta})$ . Furthermore, it can be easily seen that  $V(\mathbf{R}, \lambda, \mu, \eta) \rightarrow \infty$ , when  $\|(\mathbf{R}, \lambda, \mu, \eta) - (\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta})\| \rightarrow \infty$ .

Considering the KKT conditions of **P1-1**, we have  $\dot{V} \leq 0$  for all  $(\mathbf{R}, \lambda, \mu, \eta)$  and with equality if and only if  $(\mathbf{R}, \lambda, \mu, \eta) = (\hat{\mathbf{R}}, \hat{\lambda}, \hat{\mu}, \hat{\eta})$ .

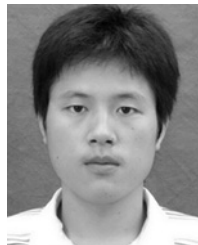
### APPENDIX C PROOF OF PROPOSITION 3

To prove the globally asymptotical stability of the high-level dynamic system, we can construct the Lyapunov function in a similar way as in the proof of Proposition 2 and apply the KKT conditions to verify that all the three conditions in Lyapunov's Theorem are satisfied. Here, the detailed proof is omitted with limited space.

### REFERENCES

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

- [2] H. Schwarz, D. Marpe, and T. Wiegand, "Overview of the scalable video coding extension of the H.264/AVC standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1103–1120, Sep. 2007.
- [3] A. Zemlianov and G. de Veciana, "Capacity of ad hoc wireless networks with infrastructure support," *IEEE J. Sel. Areas Commun.*, vol. 23, no. 3, pp. 657–667, Mar. 2005.
- [4] B. Liu, Z. Liu, and D. Towsley, "On the capacity of hybrid wireless networks," in *Proc. IEEE INFOCOM*, vol. 2, Mar.–Apr. 2003, pp. 1543–1552.
- [5] U. C. Kozat and L. Tassiulas, "Throughput capacity of random ad hoc networks with infrastructure support," in *Proc. MobiCom*, 2003, pp. 55–65.
- [6] 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.
- [7] T. Ho, R. Koetter, M. Médard, D. R. Karger, and M. Effros, "The benefits of coding over routing in a randomized setting," in *Proc. IEEE Int. Symp. Inform. Theory*, Jun.–Jul. 2003, p. 442.
- [8] L. Chen, T. Ho, S. H. Low, M. Chiang, and J. C. Doyle, "Optimization based rate control for multicast with network coding," in *Proc. IEEE INFOCOM*, May 2007, pp. 1163–1171.
- [9] X. Zhu, T. Schierl, T. Wiegand, and B. Girod, "Video multicast over wireless mesh networks with scalable video coding (SVC)," in *Proc. VVIP*, Jan. 2008, pp. 682205-1–682205-8.
- [10] J. Yan, K. Katrinis, M. May, and B. Plattner, "Media and TCP-friendly congestion control for scalable video streams," *IEEE Trans. Multimedia*, vol. 8, no. 2, pp. 196–206, Apr. 2006.
- [11] J. Zhao, F. Yang, Q. Zhang, Z. Zhang, and F. Zhang, "LION: Layered overlay multicast with network coding," *IEEE Trans. Multimedia*, vol. 8, no. 5, pp. 1021–1032, Oct. 2006.
- [12] J. Zou, H. Xiong, C. Li, L. Song, Z. He, and T. Chen, "Prioritized flow optimization with multi-path and network coding based routing for scalable multirate multicasting," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 21, no. 3, pp. 259–273, Mar. 2011.
- [13] D. S. Lun, M. Médard, and R. Koetter, "Efficient operation of wireless packet networks using network coding," in *Proc. IWCT*, 2005, pp. 1–5.
- [14] R. Koetter and M. Médard, "An algebraic approach to network coding," *IEEE/ACM Trans. Networking*, vol. 11, no. 5, pp. 782–795, Oct. 2003.
- [15] A. F. Dana, R. Gowaikar, R. Palanki, B. Hassibi, and M. Effros, "Capacity of wireless erasure networks," *IEEE Trans. Inform. Theory*, vol. 52, no. 3, pp. 789–804, Mar. 2006.
- [16] D. S. Lun, M. Médard, and M. Effros, "On coding for reliable communication over packet networks," in *Proc. Allerton Conf. Commun. Control Comput.*, 2004, pp. 20–29.
- [17] P. A. Chou, Y. Wu, and K. Jain, "Practical network coding," in *Proc. Allerton Conf. Commun. Control Comput.*, 2003, pp. 40–49.
- [18] R. W. Yeung, "Multilevel diversity coding with distortion," *IEEE Trans. Inform. Theory*, vol. 41, no. 2, pp. 412–422, Mar. 1995.
- [19] K. Yuen, B. Liang, and B. Li, "A distributed framework for correlated data gathering in sensor network," *IEEE Trans. Vehicular Technol.*, vol. 57, no. 1, pp. 578–593, Jan. 2008.
- [20] P. Gupta and P. R. Kumar, "The capacity of wireless networks," *IEEE Trans. Inform. Theory*, vol. 46, no. 2, pp. 388–404, Mar. 2000.
- [21] K. Stuhlmueller, N. Farber, M. Link, and B. Girod, "Analysis of video transmission over lossy channels," *IEEE J. Sel. Areas Commun.*, vol. 18, no. 6, pp. 1012–1032, Jun. 2000.
- [22] D. S. Lun, N. Ratnakar, M. Médard, R. Koetter, D. R. Karger, T. Ho, and E. Ahmed, "Minimum-cost multicast over coded packet networks," *IEEE Trans. Inform. Theory*, vol. 52, no. 6, pp. 2608–2623, Jun. 2006.
- [23] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
- [24] D. Palomar and M. Chiang, "A tutorial on decomposition methods and distributed network resource allocation," *IEEE J. Sel. Areas Commun.*, vol. 24, no. 8, pp. 1439–1451, Aug. 2006.
- [25] D. P. Bertsekas, *Nonlinear Programming*, 2nd ed. Belmont, MA: Athena Scientific, 1999.
- [26] S. Shakkottai and R. Srikant, *Network Optimization and Control* (Foundations and Trends in Networking, vol. 2, no. 3). Boston, MA/Delft, The Netherlands: Now Publishers, 2007.
- [27] A. Nedic and D. P. Bertsekas, "Incremental subgradient methods for nondifferentiable optimization," *SIAM J. Optimization*, vol. 12, no. 1, pp. 109–138, 2001.
- [28] M. Dai, D. Loguinov, and H. M. Radha, "Rate-distortion analysis and quality control in scalable internet streaming," *IEEE Trans. Multimedia*, vol. 8, no. 6, pp. 1135–1146, Dec. 2006.
- [29] H. Khalil, *Nonlinear Systems*, 2nd ed. Upper Saddle River, NJ: Prentice-Hall, 1996.
- [30] I. Cidon, R. Rom, and Y. Shavitt, "Analysis of multi-path routing," *IEEE/ACM Trans. Netw.*, vol. 7, no. 6, pp. 885–896, Dec. 1999.
- [31] R. Ogier, V. Rutenburg, and N. Shacham, "Distributed algorithms for computing shortest pairs of disjoint paths," *IEEE Trans. Inform. Theory*, vol. 39, no. 2, pp. 443–455, Mar. 1993.
- [32] *The Network Simulator NS-2* [Online]. Available: <http://www.isi.edu/nsnam/ns>
- [33] *JSVM 1.0 Software*, ISO/IEC JTC1/SC29/WG11/N6900, Hong Kong, China, Jan. 2005.
- [34] P. Amon, T. Rathgen, and D. Singer, "File format for scalable video coding," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1174–1185, Sep. 2007.
- [35] B. Hamdaoui and P. Ramanathan, "Rate feasibility under medium access contention constraints," in *Proc. IEEE GLOBECOM*, Dec. 2003, pp. 3020–3024.
- [36] X. Zhu, P. Agrawal, J. P. Singh, T. Alpcan, and B. Girod, "Distributed rate allocation policies for multi-homed video streaming over heterogeneous access networks," *IEEE Trans. Multimedia*, vol. 11, no. 4, pp. 752–764, Jun. 2009.
- [37] S. Deb and R. Srikant, "Congestion control for fair resource allocation in networks with multicast flows," *IEEE/ACM Trans. Netw.*, vol. 12, no. 2, pp. 274–285, Apr. 2004.
- [38] Y.-K. Wang, M. M. Hannuksela, S. Pateux, A. Eleftheriadis, and S. Wenger, "System and transport interface of SVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 17, no. 9, pp. 1149–1163, Sep. 2007.
- [39] K. Kar, S. Sarkar, and L. Tassiulas, "A scalable low overhead rate control algorithm for multirate multicast sessions," *IEEE J. Sel. Areas Commun.*, vol. 20, no. 8, pp. 1541–1557, Oct. 2002.
- [40] K. Kar, S. Sarkar, and L. Tassiulas, "Optimization based rate control for multirate multicast sessions," in *Proc. IEEE INFOCOM*, Apr. 2001, pp. 123–132.
- [41] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, no. 9, pp. 1253–1265, Sep. 1960.
- [42] E. O. Elliott, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, pp. 1977–1997, Sep. 1963.
- [43] C. Wu and B. Li, "Optimal rate allocation in overlay content distribution," in *Proc. IFIP Netw.*, LNCS 4479, May 2007, pp. 678–690.



**Chenglin Li** received the B.S. and M.S. degrees in electronic engineering from Shanghai Jiao Tong University, Shanghai, China, in 2007 and 2009, respectively. He is currently working toward the Ph.D. degree from the Department of Electronic Engineering, Shanghai Jiao Tong University.

His current research interests include network oriented image/video processing and communication, and network based optimization for video sources.



**Hongkai Xiong** (M'01–SM'10) received the Ph.D. degree in communication and information systems from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 2003.

Since 2003, he has been with the Department of Electronic Engineering, SJTU, where he is currently an Associate Professor. From December 2007 to December 2008, he was with the Department of Electrical and Computer Engineering, Carnegie Mellon University, Pittsburgh, PA, as a Research Scholar. He has published over 70 international journal/conference papers. In SJTU, he directs the Intelligent Video Modeling Laboratory and the multimedia communication area in the Key Laboratory of the Ministry of Education of China, Shanghai—"Intelligent Computing and Intelligent System" which is also co-granted by Microsoft Research, Shanghai. His current research interests include source coding/network information theory, signal processing, computer vision and graphics, and statistical machine learning.

Dr. Xiong was the recipient of New Century Excellent Talents in University in 2009. In 2008, he received the Young Scholar Award of Shanghai Jiao Tong University. He has served on various IEEE conferences as a technical program committee member. He is a member of the Technical Committee on Signal Processing of the Shanghai Institute of Electronics.



**Junni Zou** (M'06) received the M.S. and Ph.D. degrees in communication and information systems from Shanghai University, Shanghai, China, in 2004 and 2006, respectively.

Since 2006, she has been with the School of Communication and Information Engineering, Shanghai University, where she is currently an Associate Professor. She has published over 30 international journal/conference papers. Her current research interests include distributed resource allocation, multimedia communication, and network information theory.

Dr. Zou acts as a member of the Technical Committee on Signal Processing of the Shanghai Institute of Electronics.



**Zhihai He** (S'98–M'01–SM'06) received the B.S. degree in mathematics from Beijing Normal University, Beijing, China, in 1994, the M.S. degree in mathematics from the Institute of Computational Mathematics, Chinese Academy of Sciences, Beijing, in 1997, and the Ph.D. degree in electrical engineering from the University of California, Santa Barbara, in 2001.

In 2001, he joined Sarnoff Corporation, Princeton, NJ, as a Technical Staff Member. In 2003, he joined the Department of Electrical and Computer Engineering, University of Missouri, Columbia, as an Assistant Professor. His current research interests include image/video processing and compression, network transmission, wireless communication, computer vision analysis, sensor networks, and embedded system design.

Dr. received the 2002 IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY Best Paper Award, and the SPIE VCIP Young Investigator Award in 2004. Currently, he is an Associate Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY and the *Journal of Visual Communication and Image Representation*. He is a Guest Editor for the IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY special issue on video surveillance. He is a member of the Visual Signal Processing and Communication Technical Committee of the IEEE Circuits and Systems Society. He is a technical program committee member or the session chair for a number of international conferences.