A distributed algorithm for inter-layer network coding-based multimedia multicast in Internet of Things

Junyi Wang\textsuperscript{a,b,*}, Zhihui Liu\textsuperscript{c}, Yuxiang Shen\textsuperscript{a}, Hongbin Chen\textsuperscript{a,d}, Lin Zheng\textsuperscript{d}, Hongbing Qiu\textsuperscript{d}, Shi Shue\textsuperscript{e}

\textsuperscript{a} Guangxi Experiment Center of Information Science, Guilin 541004, China
\textsuperscript{b} Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin 541004, China
\textsuperscript{c} Key Lab of Universal Wireless Communications, Ministry of Education, Beijing University of Posts and Telecommunications, Beijing 100876, China
\textsuperscript{d} Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China
\textsuperscript{e} School of Mathematics and Computational Science, Xiangtan University, Xiangtan 411005, China

\textbf{ARTICLE INFO}

\textbf{Article history:}
Received 15 May 2014
Revised 22 November 2015
Accepted 25 November 2015
Available online xxx

\textbf{Keywords:}
Internet of Things
Multimedia multicast
Inter-layer network coding
Flow optimization
Distributed algorithm

\textbf{ABSTRACT}

With the development of the Internet of Things (IoT), more and more devices that have diverse computing and communication capabilities are interconnected. For multimedia multicast in the IoT, a fixed flow rate cannot meet the quality-of-service requirements of heterogeneous devices and each device may not get the information from all other devices. In order to satisfy the heterogeneous requirements, we develop a distributed algorithm to solve the inter-layer flow optimization problem based on network coding for multimedia multicast in the IoT, by using primal decomposition and primal-dual analysis. We also apply Lyapunov theory to prove the convergence and global stability of the proposed iterative algorithm. Numerical results demonstrate that the proposed algorithm has better flexibility, stability, and implementation advantages than the previous intra-layer ones.

© 2015 Elsevier Ltd. All rights reserved.

1. Introduction

Traditional internet has connected billions of computers together, through which people can exchange information conveniently. Today, in the era of the Internet of Things (IoT), machine themselves communicate with other machine or human. Although there is no standard identification of the IoT, \cite{1} provided a good description as follows: "Things have identities and virtual personalities operating in smart spaces using intelligent interfaces to connect and communicate within social, environment, and user contexts". Around us, a lot of "things", such as sensors, actuators, and smart phones are connected to create a ubiquitous communication environment. The "things" can realize some communication functions by discovering their neighbors to cooperate with them \cite{1,2}.

Multimedia communication in the IoT is a very hot topic, since the "things" are creating, transmitting, and receiving multimedia information everywhere. However, the devices such as smart phones, sensors, and personal computers are different from

Reviews processed and recommended for publication to the Editor-in-Chief by Guest Editor Dr. Lingjia Liu.

* Corresponding author at: Key Laboratory of Cognitive Radio and Information Processing, Guilin University of Electronic Technology, Guilin 541004, China.
Tel.: +86 18107732588.

E-mail addresses: wangjy@guet.edu.cn (J. Wang), liuzhihui1225@163.com (Z. Liu), 298087702@qq.com (Y. Shen), chbscut@guet.edu.cn (H. Chen), cnzlin@163.com (L. Zheng), qihb@guet.edu.cn (H. Qiu), shushi@xtu.edu.cn (S. Shu).

http://dx.doi.org/10.1016/j.compeleceng.2015.11.026
0045-7906/© 2015 Elsevier Ltd. All rights reserved.

Please cite this article as: J. Wang et al., A distributed algorithm for inter-layer network coding-based multimedia multicast in Internet of Things, Computers and Electrical Engineering (2015), http://dx.doi.org/10.1016/j.compeleceng.2015.11.026
each other in the capabilities of processing and the rate of receiving. So if the multimedia information is transmitted at a fixed rate, it may exceed the bandwidth budget of some devices and the quality-of-service requirements of some devices may not be satisfied.

Two kinds of quality-of-service requirements for multimedia services in heterogeneous networks were proposed in [3]: first, it must be able to effectively use all the available bandwidth; second, it must control the place of computation. Therefore, the first question of multimedia communication is how to make full use of the available bandwidth. One method to save bandwidth is to use the multicast scheme. In a multicast process, the receivers are divided into several multicast groups based on their requirements. For the heterogeneous receivers in the context of the IoT, multi-rate multicast is preferred, because different devices in the same multicast group could request different quality-of-service [4].

Network coding is an effective technology that can improve the throughput. It extends the functionality of nodes from storing/forwarding packets to performing algebraic operations on the received data [5]. A large number of studies have shown that network coding can dramatically improve the throughput of traditional routing, the transmission efficiency of information, etc [5–7].

Naturally, people has considered to combine the multi-rate multicast scheme with network coding. For the multi-rate multicast problem, most researchers consider the situation that flow rates for each multicast session or the link capacities are fixed. However, for the desired data rates of the receivers, the construction of optimal multi-rate multicast network codes is sometimes NP-hard [8], or may even not exist [9]. In such cases, flow control is needed to make full use of bandwidth while avoiding congestion and maintaining certain fairness among the competing flows in the network. For flow rate control, at present some researchers have proposed layered multicast schemes [10,11] applying network coding to improve performance of layered multicast based on traditional routing [12]. In general, they divide the network into different layers and construct single-rate multicast network coding for each layer [13,14]. However, these methods only apply network coding inside one layer, therefore cannot realize the full potential of network coding.

The second question of multimedia communication is where to perform the computation. In the context of the IoT, the number of intermediate nodes and sink nodes are increasing at a very fast rate. Any node in the multimedia transmission network does not know the capabilities of all other nodes, and even the network topology itself is varying from time to time. The authors in [15] suggested that “the computational part can be developed using distributed processing or cloud computing”. The inter-layer network coding scheme in [16,17] has greater flexibility in optimizing the data flow compared to previous methods, and thus achieves higher throughput. But this scheme cannot be applied in the context of IoT, because it is not a distributed algorithm. Therefore, in this paper, we study the distributed flow rate control problem for multimedia multicast based on inter-layer network coding in the IoT.

The main contributions of this paper are summarized as follows:

• We study a method to save bandwidth and perform the computation efficiently for multimedia multicast in the IoT.

• We introduce the concept of transfer price specially for inter-layer network coding-based multicast flows, which is a Lagrange multiplier in the proposed algorithm. It, together with congestion price proposed in [16,17], are buffered in virtual queues at each node to determine the flow control strategy. In the scheme proposed in [16,17], the interactions of transfer price and congestion price emerged from multicast flows and the inverse of the derivative function of utility are not well expressed. We adopt an alternative iteration method which has been successfully used in [11] and extend it to inter-layer network coding-based rate control relying on current transfer price and congestion price. Therefore the problems encountered in rate control, which was solved by the direct solving method for intra-layer network coding [13], are avoided.

• We propose a distributed algorithm of joint rate control and multicast layer flow allocation. We use a partially-primal and dual gradient-based approach complies with the adaptive control of signal transmission mechanism to solve the optimization problem, while reducing the required control information. Furthermore, we mathematically proves the convergence and the globally stability of the proposed algorithm for the network coding-based multi-rate multicast, by using Lyapunov theory.

The rest of this paper is organized as follows. Section 2 briefly discusses the related works. Section 3 describes the details of the system model and the formulation of the flow optimization problem for multi-rate multicast based on inter-layer network coding. Section 4 proposes the distributed algorithm for the flow optimization problem and proves that the proposed algorithm is globally stable. In Section 5, numerical examples are provided to illustrate the convergence, performance improvement, and implementation advantages of the proposed algorithm. Finally, Section 6 concludes the paper.

2. Related works

In recent years, some schemes about network coding-based layered multicast have been proposed to address the problems emerging in the field of multi-rate multicast. Compared to traditional routing, these schemes had substantially improved the throughput [18]. Based on the traditional layered multicast scheme [12], Zhao et al. [10] proposed a layered overlay multicast scheme to cope with heterogeneity and to improve throughput. Zou et al. [14] studied the prioritized flow optimization for scalable video coding and multicast over networks, jointly considering multi-path video streaming, network coding-based routing, and network flow control. Lakshminarayana and Eryilmaz [11] adopted layered multicast mechanism with intra-layer network coding and nested optimization approach to investigate the multi-rate multicast utility maximization problem.
However, many layered multicast schemes [19] that mainly used intra-layer network coding divided the multicast network into multiple multicast layers according to some standard. Each multicast layer can be seen as a session and network coding is used in each layer, respectively. The existing polynomial-time algorithm [7] was used respectively for each layer, and then the superposition approach was used over multiple layers [20].

Intuitively, the main feature of intra-layer network coding is that network coding is performed only inside one layer, which cannot achieve the full potential of network coding. So the researchers are focusing on the promising inter-layer network coding technique recently. The authors in [16,17] focused on network code construction for novel layered multicast using inter-layer network coding and provide a concentrating solution for flow optimization. Widmer et al. [21] provided a heuristic algorithm for rate allocation and code assignment based on layered multicast streaming with inter-layer network coding. Owing to these literature mainly concentrating on coding strategies, the corresponding solutions are in view of centralized control. This paper studies distributed flow control based on layered multicast with inter-layer network coding. Our scheme allows flow in multicast layer to carry data in all data layers 1, 2, ..., k. Thus, messages sent in layer k are linear combinations of data in layers 1, 2, ..., k. While [16,17] provided no optimization methods for inter-layer network coding, this paper provides a dynamic, distributed flow control algorithm for the inter-layer network coding scheme. This distributed flow control algorithm use a partially-primal and dual decomposition method to get the dynamic solution which is obtained by just using local information to maximize the users’ utilities. As for the dual variables in the Lagrange multiplier method, except for the concrete congestion price proposed in [16,17], we introduce the transfer price. The price is for some part of the flow in lower layers to be transmitted in higher layers when we use the inter-layer network coding technique. The part of flow that brings the congestion in lower layers will be transferred to the higher layers, so it needs to pay the price for buffering the transferred data.

Furthermore, to analyze the stability and convergence of the proposed algorithm, we adopt classical Lyapunov theory [22], which is triumphantly devoted in the convergence analysis of control algorithms, such as stability analysis of iterative algorithms for scalable multi-rate multicast [14], congestion control for single-rate multicast flows with network coding [13] and multi-rate multicast with intra-layer network coding [11], etc.

3. System model and problem formulation

3.1. Network model

First, we build a optimization model of the flow routing based on network coding to make full use of available bandwidth.

We consider a network, denoted by a graph $G = (V,E)$, with a set $V$ of nodes and a set $E$ of links. Meanwhile, the set of sink nodes is denoted by $D$. Let the receiving rate of the sink node $d \in D$ denoted by $R(d)$. For each node $v \in V$, we use $ln(v)$ to represent the set of all incoming links to node $v$ and $Out(v)$ to represent the set of all outgoing links from node $v$. The link belonging to set $E$ is denoted by the directed pair $(i, j)$ of the nodes $i, j \in V$ that it connects. Assume the multicast source sends the messages $a_1, \ldots, a_S$, of equal size, and each source packet $a_i (1 \leq i \leq S)$ is treated as an indivisible flow unit.

3.2. Inter-layer network coding scheme and flow optimization

First of all, implementation steps of the inter-layer network coding scheme in [16,17] are summarized as follows:

1. According to each sink’s max-flow value [6] and utility function, partition $D$ into several subsets $D_1, D_2, \ldots, D_N$ (which means that there are $N$ multicast groups) such that: $\forall d \in D_k, d' \in D_{k+1}, R(d) < R(d')$, and $\forall d, d' \in D_k, R(d) = R(d') \equiv R_k$ and $U_k(R_k) = U_{d'}(R_k)$, where $U_k(\cdot)$ represents the utility function of sink $d$.

2. Based on the $N$ multicast groups determined by (1), the source data flow is encoded into a series of non-overlapping layers $L_1, L_2, \ldots, L_N$.

3. With the idea of inter-layer network coding, the source node or the intermediate nodes could choose to transport the combinations of all the data in the first $k$ data layers to sink $d$. That means some data is allowed to be transferred from one layer to a higher layer at the source node or some intermediate nodes. We say that some part of the flow in lower layers is allowed to be “delayed” and transmitted in higher layers. However, finally all the “delayed” flow have to reach sink $d$ in order to ensure that $d$ receives all the data in the first $k$ data layers.

Then, according to the specific inter-layer network coding scheme discussed above, the flow optimization problem $P$ can be formulated as follows:

\[
\text{max} \sum_{d \in D} U_d(R_k(d)) \quad (1a)
\]

subject to:

\[
b_{j}^{d}l = \sum_{(i,j) \in ln(j)} x_{i,j}^{d,l} - \sum_{(j, l) \in Out(j)} x_{j,l}^{d,l}, \forall j \in V, d \in D, 1 \leq l \leq N \quad (1b)
\]

\[
R_j \geq -\sum_{i=1}^{j} b_{i}^{d,l}, \forall d \in D, 1 \leq j < L(d) \quad (1c)
\]

\[
R_j = -\sum_{i=1}^{j} b_{i}^{d,l}, \forall d \in D, j = L(d) \quad (1d)
\]
4.1. Distributed algorithm

4. Distributed algorithm for flow optimization


\[
\sum_{i=1}^{j} b_{n}^{d,l} \geq 0, \quad \forall \ d \in D, \ 1 \leq j < L(d), \ n \notin \{s, d\} \tag{1e}
\]

\[
\sum_{i=1}^{j} b_{n}^{d,l} = 0, \quad \forall \ d \in D, \ j = L(d), \ n \notin \{s, d\} \tag{1f}
\]

\[
-j \sum_{i=1}^{j} b_{s}^{d,l} \geq \sum_{i=1}^{j} b_{d}^{d,l}, \quad \forall \ d \in D, \ 1 \leq j < L(d) \tag{1g}
\]

\[
-j \sum_{i=1}^{j} b_{s}^{d,l} = \sum_{i=1}^{j} b_{d}^{d,l}, \quad \forall \ d \in D, \ j = L(d) \tag{1h}
\]

\[
y_{l,j}^{d,l} \geq x_{l,j}^{d,l}, \quad \forall \ d \in D, \ 1 \leq l \leq N \tag{1i}
\]

\[
\sum_{l=1}^{N} y_{l,j}^{d,l} = C_{l,j}, \quad \forall (i, j) \in E \tag{1j}
\]

where \(U_{d}(\cdot)\), as a function of rate \(R(d)\), is assumed to be continuously differentiable, increasing, and strictly concave for the flow rate. \(L(d)\) denotes the number of the data layers that sink \(d\) will receive (i.e. \(L(d) = k\) if and only if \(d \in D_{k}\)). Let \(x_{l,j}^{d,l}\) be the flow on link \((i, j)\) for sink \(d\) in layer \(l\). Define \(b_{s}^{d,l}\) to be the potential of node \(j\) for sink \(d\) in layer \(l\), which is the difference between the incoming flow and the outgoing flow. Let \(y_{l,j}^{d,l}\) be the actual flow on link \((i, j)\) in layer \(l\) (over all sinks). \(C_{l,j}\) denotes the capacity of each link \((i, j) \in E\) in the network.

In the optimization problem \(P\), constraints (1c), (1e), and (1g) concern the flow transfer at the source node, the intermediate nodes, and the sink nodes, respectively. Constraints (1d), (1f), and (1h) are the flow balance constraints at the source node, the intermediate nodes, and the sink nodes, respectively. Constraint (1i) is the network coding constraint. Constraint (1j) confines that the actual flow in each link cannot exceed the capacity of that link.

4. Distributed algorithm for flow optimization

As the authors in [16,17] did not give optimization algorithms for the inter-layer network coding scheme, in this section we use primal decomposition and primal-dual analysis to form a distributed algorithm for the flow optimization problem \(P\). This algorithm uses only local information of the nodes in the network, so as to reduce the required control information and comply with the adaptive control of signal transmission mechanism. Moreover, this algorithm can make each node collaborated with the user terminals and let all of them interconnect efficiently to improve the communication performance of multi-rate multicast. In addition, we prove that the proposed algorithm is globally stable.

4.1. Distributed algorithm

First, we find that Problem \(P\) can be divided into two subproblems: Subproblem \(P_1\) is \(\max \sum_{d \in D} U_{d}(R_{d}(d))\) with respect to (1c)–(1h), and Subproblem \(P_2\) is \(\max L(R, p, v)\) with respect to (1b), (1i), and (1j), where \(L(R, p, v)\) is the Lagrangian of the Problem \(P\) with respect to the flow balance and flow transfer constraints, that is

\[
L(R, p, v) = \sum_{d \in D} U_{d}(R_{d}(d)) - \sum_{d \in D} \sum_{1 \leq j < L(d)} p_{d}^{j} \left( -R_{j} - \sum_{i=1}^{j} b_{s}^{d,l} \right) - \sum_{d \in D} \sum_{n \notin \{s, d\}} \sum_{1 \leq j < L(d)} p_{n}^{d,l} \left( -j b_{n}^{d,l} \right) - \sum_{d \in D} \sum_{1 \leq j < L(d)} p_{d}^{l} \left( b_{d}^{d,l} \right) - \sum_{d \in D} \sum_{n \notin \{s, d\}} \sum_{1 \leq j < L(d)} v_{d}^{l} \left( R_{l}(d) + \sum_{i=1}^{l(d)} b_{d}^{d,l} \right) \tag{2}
\]

where vector \(R = \{R_{d}(d), d \in D\}\), vector \(v = \{v_{n}^{l(d)}, d \in D, n \in V\}\), and vector \(p = \{p_{d}^{j,l}, d \in D, n \in V, 1 \leq j < L(d)\}\). Besides, define vector \(x = \{x_{l,j}^{d,l}, (i, j) \in E, d \in D, 1 \leq l \leq N\}\), and vector \(y = \{y_{l,j}^{d,l}, (i, j) \in E, 1 \leq l \leq N\}\).

For each node \(n \in V\) and \(1 \leq j < L(d)\), \(p_{d}^{j,l}\) can be interpreted as the “transfer price” at node \(n\) for multicast layer \(j\) and sink node \(d \in D\), and \(v_{n}^{l(d)}\) can be interpreted as the “congestion price” at node \(n\) for sink \(d \in D\). \(p_{n}^{d,l}\) means that, due to its insufficient memory or slow processor, node \(n\) needs to pay the price for failing to make the flow designated to sink \(d\) queuing in its buffer, updating the routing tables, and so on; \(p_{d}^{j,l}\) means that the cache overhead is needed when flow at node \(n\) is transferred from layer \(j\) to a higher layer and the node \(n\) may transfer the congested flow in layer \(j\) to a higher layer because of the flow transfer.

Please cite this article as: J. Wang et al., A distributed algorithm for inter-layer network coding-based multimedia multicast in Internet of Things, Computers and Electrical Engineering (2015), http://dx.doi.org/10.1016/j.compeleceng.2015.11.026
Again, the Lagrange function $L(R, p, v)$ is separable: $L(R, p, v) = L(R) + L(p, v)$, where

$$L(R) = \sum_{d \in D} U_d(R_{L(d)}) + \sum_{d \in D} \sum_{1 \leq j \leq n} p_{d,j}^d R_j - \sum_{d \in D} v_{d}^{d,L(d)} R_{L(d)}$$

and

$$L(p, v) = \sum_{d \in D} \sum_{1 \leq j \leq n} p_{d,j}^d \left( \sum_{i=1}^{j} b_{d,i}^j \right) + \sum_{d \in D} \sum_{n \in \{d,s\}} \sum_{1 \leq j \leq n} p_{d,j}^d \left( \sum_{i=1}^{j} b_{d,i}^j \right) - \sum_{d \in D} v_{d}^{L(d)} \left( \sum_{i=1}^{L(d)} b_{d,i}^j \right)$$

Thus the problem $P_1$ can be divided into two parts: $P_{11}$ and $P_{12}$, where $P_{11}$ is max $L(R)$, and $P_{12}$ is max $L(p, v)$.

Then we obtain the following joint rate control and multicasting layer flow allocation algorithm as follows: for fixed vector $v$ and vector $p$, to solve Subproblem $P_{11}$; then for fixed vector $x$, vector $y$, and obtained vector $R$, to solve Subproblem $P_{12}$; for obtained vector $p$, vector $v$, and vector $R$, to solve Subproblem $P_2$ at last.

4.1. Flow rate control

At time $t$, given congestion price vector $v(t)$ and transfer price vector $p(t)$, source node $s$ adjusts its sending rate $R_k$ to the sink/sinks $d$ in layer $k$ (equal to $L(d)$) according to the total net congestion price $\overline{p}_{s,k}(t)$ of layer $k$ which is generated locally at the source node, namely, $\forall d \in D$,

$$R_k(t) = \left[ R_k(t) + \alpha_k(t) \left( \sum_{d \in D_k} U_{d,k}(R_k(t-1)) - \overline{P}_{s,k}(t) \right) \right]^{\varphi_1}_{\varphi_0},$$

(3)

where $U_{d,k}(R) = \frac{dU_{d,k}(R)}{dR}; \overline{P}_{s,k}(t) = \sum_{d \in D_k} \overline{p}_{s,d,k}(t) - \sum_{d' \in \partial_d^{-},d' \neq k} \overline{p}_{s,d',k}(t)$; and $\overline{p}_{s,d,k}(t)$ [11] denotes the projection of $z$ onto the interval $[a,b]$. Furthermore, $\alpha_k$ is a small positive step-size parameter, $0 < \varphi_0 < \min_k R_k^*$ where $R^*$ is the optimal solution, and $\varphi_1$ is a finite constant that is greater than $\sum_{(i,j) \in E} C_{i,j}$.

4.1.2. Price update

This section will provide the mechanism of updating the transfer price and congestion price. According to the projection method in [11], by solving Subproblem $P_{12}$, the transfer price and congestion price is updated as follows:

(i) Each node $h$ belonging to set $V$ updates its transfer price with respect to sink $d \in D_k (1 \leq k \leq N)$ and multicasting layer $j(1 \leq j \leq 1)$, according to

$$p_{d,j}^h(t+1) = p_{d,j}^h(t) + \gamma_s(t) \left[ \sum_{i=1}^{j} \left( \sum_{(s,h) \in \text{Out} (s)} x_{d,i}^h \right) - R_j \right]_{p_{d,j}^h}$$

(ii) Node $h \in V$ updates its transfer price with respect to multicast layer $k$ and sink $d \in D_k$ where $1 \leq k \leq N$ follows

$$v_{d}^{d,k}(t+1) = v_{d}^{d,k}(t) + \delta_t \left( R_k - \sum_{i=1}^{k} \left( \sum_{(s,h) \in \text{Out} (s)} x_{d,i}^h \right) \right)_{v_{d}^{d,k}}$$

(4)

for each sink $\forall d \in D_k (1 \leq k \leq N), 1 \leq j \leq k-1$, where both step size $\gamma_s(t)$ and step size $\gamma_n(t)$ are positive, and $n \in V \backslash \{s,d\}$.

(ii) Node $h \in V$ updates its transfer price with respect to multicast layer $k$ and sink $d \in D_k$ where $1 \leq k \leq N$ follows

$$v_{d}^{d,k}(t+1) = v_{d}^{d,k}(t) + \delta_t \left( R_k - \sum_{i=1}^{k} \left( \sum_{(s,h) \in \text{Out} (s)} x_{d,i}^h \right) \right)_{v_{d}^{d,k}}$$

(5)

for each sink $\forall d \in D_k$ and $1 \leq k \leq N$, where both step size $\gamma_s(t)$ and step size $\gamma_n(t)$ are positive, and $n \in V \backslash \{s,d\}$.

Please cite this article as: J. Wang et al., A distributed algorithm for inter-layer network coding-based multimedia multicast in Internet of Things, Computers and Electrical Engineering (2015), http://dx.doi.org/10.1016/j.compeleceng.2015.11.026
Note that the transfer price identically equal to zero in other cases which mean there is no transfer. For example, if \( d \in D_2 \), then \( p^{d,3} = 0 \), and that is because the sinks in the layer 2 have completed the reception of all data required during the first two layers. Another example is \( p^{d,2} = 0 \) which derives from the fact that the flow designated to the sinks belonging to layer 2 in the layer 2 can no longer be transferred to a higher layer.

4.1.3. Multicast layer flow allocation

Intuitively, multicast data layer with higher link net congestion price should be allocated more link bandwidth. We elect the layers of which the flow will aggravate the link net congestion and the layers of which the flow have caused more serious congestion on the link, and take the average cumulative congestion of these layers as the threshold value to update the link bandwidth.

The Subproblem \( P_2 \) can be solved by using the subgradient method [23]. Specifically, let \( \hat{P}^{j, k(d)}_{h,n} = P^{d,k}_{h,n} - P^{d,j}_{h,n} \), where \((h, n) \in E, d \in D_k\) and \( i \leq k \leq N \), and interpret \( \hat{P}^{j, k(d)}_{h,n} \) as the “net congestion price” at link \((h, n)\) for the flow of layer \( i \) destined to sink \( d \) which belong to the multicast layer \( k \). At each link \((h, n)\), the amount of bandwidth \( y^i_{h,n} \) that is allocated to layer \( i \) is as follows:

\[
y^i_{h,n}(t + 1) = y^i_{h,n}(t) + \varepsilon_{h,n}(t)\left[ w^i_{h,n} - E_{h,n}[w]\right]^{+}, \tag{6}
\]

where

\[
V^{d,k}_{h,n} = \begin{cases} 
    v^d_h - v^k_h - v^d_k, & \text{if } h = s, n \notin \{d, s\}, \\
    v^d_h + v^k_h, & \text{if } h \notin \{d, s\}, n = d, \\
    v^d_h, & \text{if } h = s, n = d, \\
    v^d_h - v^k_h, & \text{otherwise};
\end{cases}
\]

\[
P^{d,k}_{h,n} = \begin{cases} 
    \sum_{j=1}^{k-1} (p^d_h - p^d_j - p^d_n), & \text{if } h = s, n \notin \{d, s\}, \\
    \sum_{j=1}^{k-1} (p^d_h + p^d_j), & \text{if } h \notin \{d, s\}, n = d, \\
    \sum_{j=1}^{k-1} p^d_j, & \text{if } h = s, n = d, \\
    \sum_{j=1}^{k-1} (p^d_j - p^d_n), & \text{otherwise};
\end{cases}
\]

\[
[h]^+ = \begin{cases} 
    h, & \text{if } z > 0, \\
    \max\{0, h\}, & \text{if } z = 0;
\end{cases}
\]

the cumulative net congestion of layer \( i \) on link \((h, n)\) is

\[
w_{h,n}^{i} = \sum_{k=1}^{N} \sum_{d \in D_k} \left[ P^{k, i(d)}_{h,n} \right]^{+}, \tag{7}
\]

where \( \varepsilon_{h,n}(t) \) is a positive step-size and \( E_{h,n}[w(t)] \) denotes the minimal of those \( \underline{w}_{h,n}(t) \) at time \( t \) such that \( \underline{w}_{h,n}(t) = \frac{1}{\underline{b}_{h,n}(t)} \sum_{i \in N'_{h,n}(t)} w_{h,n}(t) \) with \( N'_{h,n}(t) := \{ i \mid y^i_{h,n}(t) > 0 \} \) or \( \underline{w}_{h,n}(t) \geq \underline{w}_{h,n}(t), i \in \{1, \ldots, N\} \) [13,24].

Over link \((h, n)\), a random linear combination [25] of the data of first \( i \) multicast layers to all sinks \( d \) such that \( \hat{P}^{i, k(d)}_{h,n} > 0 \) is sent at rate \( y^i_{h,n} \) namely,

\[
x^i_{h,n}(t) = \begin{cases} 
    y^i_{h,n}(t), & \text{if } \hat{P}^{i, k(d)}_{h,n} > 0, \text{ where } d \in D_k \text{ and } i \leq k \leq N \\
    0, & \text{otherwise}
\end{cases} \tag{8}
\]

According to (6) and the computing process of \( E_{h,n}[w(t)] \), we can come to the conclusion:

\[
\sum_{i=1}^{N} y^i_{h,n}(t) = 0. \tag{9}
\]

4.2. Implementation of distributed algorithm

To implement the proposed distributed algorithm, each link \((h, n)\), as well as each node \( h \), is regarded as a processor of a distributed computation system. Assume that the processor for link \((h, n)\) keeps track of variables \( y^i_{h,n} \) and \( x^i_{h,n} \), while the processor for node \( h \) keeps track of variables \( v^d_k \) and \( P^d_{h,n} \). The proposed distributed algorithm can be implemented as follows:

The time complexity of the process for each node is \( O(NLU) \) and the time complexity of the process for each link is \( O(N^2L^2U) \), where \( N \) is the number of intermediate nodes, \( U \) is the number of sink nodes, and \( L \) is the number of multi-rate multicast layers. Thus the time complexity of the proposed distributed algorithm is \( O(N^2L^2U) \).
Algorithm 1 Distributed algorithm.

Input:
1: $C_{h,n}$

Initialization:
2: $x^{d,i}_{h,n}(0) = 0, y^{d,i}_{h,n}(0) = \frac{C_{h,n}}{K}, \forall (h, n)$; choose $p^{d,j}_h(0) > 0, v^{d,k}_h(0) > 0, \forall h$

Iteration:
3: Each source $s$ computes $R_s(t)$ as equation (3)
4: Each node $h$ computes $v^{d,k}_h(t)$ and $p^{d,j}_h(t)$ as relations (4) and (5)
5: Each link $(h, n)$ computes $x^{d,i}_{h,n}(t)$ and $y^{d,i}_{h,n}(t)$ as follows:
6: Compute $w^{d,i}_{h,n}$ as equation (7)
7: Set $N'_h(t) = \{1, \ldots, N\}$, compute $w^{d,i}_{h,n}(t) = \frac{1}{|N'_h(t)|} \sum_{i \in N'_h(t)} w^{d,i}_{h,n}(t)$
8: while $y^{d,i}_{h,n}(t) = 0$ and $w^{d,i}_{h,n}(t) \leq W_h(t)$, do
9: $N'_h(t) = N'_h(t) \setminus \{i\}$, and $w^{d,i}_{h,n}(t) = \frac{1}{|N'_h(t)|} \sum_{i \in N'_h(t)} w^{d,i}_{h,n}(t)$
10: end while
11: $E_{h,n}(\bar{w})(t) = \bar{w}_{h,n}(t)$
12: Calculate $y^{d,i}_{h,n}(t)$ as equation (6)
13: if $d \in D_k$ and $(v^{d,k}_h - v^{d,k}_n) - \sum_{j=1}^{k-1} (p^{d,j}_h - p^{d,j}_n) > 0$
14: $x^{d,i}_{h,n}(t) = y^{d,i}_{h,n}(t)$
15: else $x^{d,i}_{h,n}(t) = 0$
16: end if
17: if $x^{d,i}_{h,n}(t) - x^{d,i}_{h,n}(t-1) < 10^{-2}$
18: terminate the Iteration
19: end if

4.3. Convergence analysis

The distributed control (3)–(6), and (8) is a partially-primal and dual algorithm. We consider the proposed algorithm as a nonlinear autonomous system and apply Lyapunov stability theorem to analyze the convergence behavior of the proposed algorithm as the proof in [22].

Because of the page limitation, we only point out the key of the proof.

If the equilibrium point $\bar{x}$ is some $\bar{x} \neq 0$, Lyapunov theorem is also true. Since in this case, we just need to consider the state vector of the system as $y = x - \bar{x}$, the results of Lyapunov theorem can be applied.

If $R^*, x^*,$ and $y^*$ are the optimal source rate and multicast layer flow allocation of Problem $P$, and $p^*$ and $v^*$ are optimal solution to the dual problem $L(p, v)$, then by Karush–Kuhn–Tucker (KKT) conditions [23], they satisfy the constraints of Problem $P$ and the following relations:

$$\sum_{d \in D_k} U'_d((R^*)_h) - \left( \sum_{d \in D_k} (v^*)_h^d \sum_{d' \in D_k, d' > d} (v^*)_h^{d'} \right) = 0, \forall k \in \{1, \ldots, N\}$$  (10)

$$(x^*)^{d,i}_{h,n} = \arg \max_{(x^{d,i}_{h,n} \in \mathbb{R}^+)} \sum_{i=1}^{N} \sum_{k=1}^{d} \sum_{d'} (x^*)^{d,i}_{h,n} (v^*)^{d,i}_{h,n} - (P^*)^{d,i}_{h,n} \sum_{i=1}^{N} \sum_{k=1}^{d} (x^*)^{d,i}_{h,n} = 0$$  (11)

$$(p^*)^{d,i}_{h,n} \geq 0, \forall d \in D_k (1 \leq k \leq N), 1 \leq j \leq k - 1, h \in V$$  (12)

$$\begin{cases} (P^*)^{d,i}_{s,h} - (R^*)_s + \sum_{i=1}^{j} \sum_{(s,d) \in \text{Out}(s)} (x^*)^{d,i}_{s,h} = 0, \\ (P^*)^{d,i}_{h,n} - \sum_{i=1}^{j} \sum_{(h,n) \in \text{In}(n)} (x^*)^{d,i}_{h,n} = 0, \\ (P^*)^{d,i}_{d} - \sum_{i=1}^{j} \sum_{(s,d) \in \text{In}(d)} (x^*)^{d,i}_{s,h} = 0. \\ \forall d \in D_k, 1 \leq k \leq N, 1 \leq j \leq k - 1, n \in V \setminus \{s, d\}. \end{cases}$$  (13)
Then if \((R^*, x^*, y^*, p^*, v^*)\) is an equilibrium point of the primal-dual algorithm proposed in (3)–(6), and (8), by forming the following Lyapunov function \(V(v, p, y)\), we can prove that the algorithm is globally stable by using Lyapunov theory.

5. Numerical results

In this section, numerical simulation results are presented to complement the analysis in previous sections, and to verify the convergence, the performance improvement, and the implementation advantages of the proposed algorithm.

5.1. Numerical example for modified butterfly graph

First of all, we conduct experiments on a simple multi-rate multicast graph adapted from the butterfly graph, as shown in Fig. 1(a). The multicast graph is assumed to be directed and each link has one unit of capacity. Assume that there is only one source node \(s_1\) and three sink nodes \(d_1, d_2,\) and \(d_3\), with the same utility \(U_d(R_L(d)) = \ln(1 + a \cdot R_L(d))\). We choose such a small, simple network topology to facilitate detailed discussions about the results.

5.1.1. Inter-layer network coding

According to the inter-layer network coding scheme, the sinks’ set \(D\) should be divided into two subsets \(D_1\) and \(D_2\) where \(d_1 \in D_1\) and \(d_2, d_3 \in D_2\). In addition, source messages should be encoded into two layers. Thus, the sink node \(d_1\) only receives the data \(a_1\) of layer 1, and sink nodes \(d_2\) and \(d_3\) receive the data \(a_1\) and \(a_2\) of layer 1 and layer 2.

5.1.2. Analytical solution

By theoretically analyzing the modified butterfly graph, we obtain the analytical solution of the layered network coding scheme, as shown in Fig. 1(a) and Table 1, where \(Y_{1\times24} = \{ y^i_j \mid (i,j) \in E, l = 1, 2, m = 1 \}\). Therein, in each tuple \((a, b; c, d; e, f)\) of Fig. 1(b), elements \(a\) and \(b\) indicate the information flow of the first layer and the second layer destined for sink \(d_1\), respectively; elements \(c\) and \(d\) indicate similarly for sink \(d_2\), and similarly for elements \(e\) and \(f\). In addition, in each tuple \((h, i)\) of Fig. 1(c), elements \(h\) and \(i\) over each link indicate the physical flow of the first layer and the second layer over the link, respectively.

5.1.3. Congestion price and transfer price

By carefully examining the analytical solution for the given multi-rate multicast network as shown in Fig. 1(d), we observe that the flow can indeed be divided into two multicast layers. It can be seen that the second multicast layer carries data in both layers, not just in layer 2. Moreover, note that sink node \(d_3\) does not receive any unit of flow in the first multicast layer, but then in order to compensate, receives two units of flow in the second multicast layer. Precisely, links \((w_1, v_1)\), \((r_1, d_2)\), and \((r_1, d_3)\)

\[\text{Fig. 1. The modified butterfly graph based on inter-layer network coding. (a) Network topology. (b) Information flow. (c) Physical flow. (d) Optimal layered solution.}\]
transmit a combination of $a_1$ and $a_2$. At the node $w_1$, the flow $a_1$ in the first layer is transferred to the second layer. We say that for sinks $d_2$ and $d_3$, the data $a_1$ in the first layer is “delayed” and transmitted together with the data in the second layer.

Then, we analyze the analytical layered solution for the given multicast network in Fig. 1(b)–(d) to further explain the specific physical significance of the congestion price and transfer price.

(a) Congestion price: At the node $u_1$, the flow from the incoming link $(s_1, u_1)$, when it is bigger than the throughput capability of the outgoing link $(u_1, w_1)$ or the processing power of node $u_1$, may cause congestion and need to pay the price for it.

(b) Transfer price: At the node $w_1$, some or all of the flow from the incoming link $(u_1, w_1)$ in the first multicast layer is transferred to the second multicast layer, thus the first multicast layer gets some gains.

5.1.4. Performance comparison of layered multicast schemes

In this subsection, we start with the convergence behavior of the proposed algorithm. Fig. 2 shows the evolution of source rates over different layers with step sizes $\alpha_k = \epsilon_{h,n} = 0.01, \delta_n = 0.005,$ and $\gamma_n$ being diminished gradually. It can be seen from the plots that the rate of layer 2 is well within 5% of their optimal values after 1500 iterations. Although the rate of layer 1 converges at a little slower rate for the gradually diminished step sizes of the transfer price. The convergence of the whole rate control algorithm is rather fast. We see that the source rates approach the corresponding optimum quickly.

We further compare the rate control solutions as shown in Fig. 3. In this figure, “Our Scheme A” denotes the proposed algorithm with step sizes $\alpha_k = \epsilon_{h,n} = 0.01, \delta_n = 0.005,$ and $\gamma_n$ being diminished gradually. “Our Scheme B” denotes the proposed algorithm with step sizes $\alpha_k = \epsilon_{h,n} = 0.01, \delta_n = \gamma_n = 0.005$. “SIFA A” and “SIFA B” come from [24] and [13], respectively. In “SIFA A”, physical and information flow iteration step sizes are equal to 0.005. We can see from the figures that compared to “SIFA A”, “Our Scheme A” can obtain much better multicast layer flow allocation result at the expense of more iterations. Meanwhile, due to the interaction of congestion price and transfer price, with the same multicast layer flow allocation and price adjustment, “our Scheme A” that sacrifices a little multicast layer flow allocation performance can achieve more effective rate control than “SIFA B”. In other words, “our Scheme A” gets a tradeoff between rate control and optimal flow allocation. Thus, “our Scheme A” is more suitable for multi-rate multicast based on inter-layer network coding to proceed rate control and multicast.

Table 1
Analytical solution of the layered network coding scheme.

<table>
<thead>
<tr>
<th>Physical flow</th>
<th>Y(1)</th>
<th>Y(2)</th>
<th>Y(3)</th>
<th>Y(4)</th>
<th>Y(5)</th>
<th>Y(6)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allocation</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Physical flow</td>
<td>Y(7)</td>
<td>Y(8)</td>
<td>Y(9)</td>
<td>Y(10)</td>
<td>Y(11)</td>
<td>Y(12)</td>
</tr>
<tr>
<td>Allocation</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Physical flow</td>
<td>Y(13)</td>
<td>Y(14)</td>
<td>Y(15)</td>
<td>Y(16)</td>
<td>Y(17)</td>
<td>Y(18)</td>
</tr>
<tr>
<td>Allocation</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Physical flow</td>
<td>Y(19)</td>
<td>Y(20)</td>
<td>Y(21)</td>
<td>Y(22)</td>
<td>Y(23)</td>
<td>Y(24)</td>
</tr>
<tr>
<td>Allocation</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
</tbody>
</table>

Fig. 2. The evolution of layers’ rates.

Please cite this article as: J. Wang et al., A distributed algorithm for inter-layer network coding-based multimedia multicast in Internet of Things, Computers and Electrical Engineering (2015), http://dx.doi.org/10.1016/j.compeleceng.2015.11.026
layer flow allocation. Moreover, “our Scheme B” is slightly better than “our Scheme A” for the simple butterfly graph with only a few sink nodes. However, as the sink nodes increases, “our Scheme A” is more flexible than “our Scheme B”. Besides, we notice that the time complexity of “SIFA A” and “SIFA B” are the same as our schemes.

Next, we investigate how the proposed algorithm performs when the network topology is random.

5.2. Numerical example for random network topology

In this section, we consider a random network topology as shown in Fig. 4, where the number next to the link is the capacity of that link.

5.2.1. Performance of the proposed algorithm

Here we use the more flexible scheme “our Scheme A” mentioned in Section 5.1.4 to solve the flow optimization problem for the random network topology. Fig. 5(a) shows the evolution results of the rates of layers. The results demonstrate that rate control using the proposed distributed algorithm approaches the maximum flow rates 2, 5, and 5.
5.2.2. Comparison with SIFA scheme

We choose the more recently proposed scheme “SIFA B” to be compared with “our Scheme A”. Fig. 5(b) shows the evolution results of the rates of layers for this scheme. From Fig. 5, we can see that although “our Scheme A” converges more slowly than “SIFA B”, “our Scheme A” is more stable than “SIFA B”, of which the rates are a little oscillating all the time.

6. Conclusion

In this paper, we have proposed a distributed algorithm for optimizing the flow rates of inter-layer network coding-based multi-rate multicast in Internet of Things. The algorithm can be implemented in the transport layer to adjust source rates and in the network layer to carry out network coding with multicast layer flow allocation, as well as congestion price and transfer price update. We also proved the stability and convergence of the proposed iterative algorithm using Lyapunov theory. Compared to the intra-layer network coding schemes, the layered multicast scheme that allows the network coding performed across different layers has better flexibility and stability in optimizing the data flow.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Grants 61261017, 61162008, 61362006; by the Foundation of Key Laboratory of Universal Wireless Communications under Grant KFKT-2014102; by the Guangxi Natural Science Foundation under Grants 2013GXNSFBA019334 and 2015GXNSFBB139007; by the Foundation of Guangxi Key Laboratory of Wireless Wideband Communication & Signal Processing under Grants GXKL0614101, GXKL061501, GXKL0614202; by the Foundation of Guangxi Experiment Center of Information Science under Grant 20130310; and by the Foundation of Key Laboratory of Cognitive Radio and Information Processing under Grant CRKL150112.

References


Junyi Wang received the M.E. degree in mathematics from Xiangtan University, China, in 2003; and the Ph.D. degree in signal and information processing from Beijing University of Posts and Telecommunications, China, in 2008. He is an Associate Professor in the School of Information and Communication, Guilin University of Electronic Technology, China.

Zhihui Liu received the B.S. degree in information and computing science from North University of China, Taiyuan, China, in 2009, and the M.S. degree in computational mathematics from Xiangtan University, Xiangtan, in 2012. She is currently working toward the Ph.D. degree from BUPT in Signal and Information Processing. Her current research interests include multicast, network coding, cognitive radio, and green communication.

Yuxiang Shen is working for the M.E. degree in electronics and communication engineering at Guilin University of Electronic Technology, China.

Hongbin Chen received the B.E. degree in electronic and information engineering from Nanjing University of Posts and Telecommunications, China, in 2004; and the Ph.D. degree in circuits and systems from South China University of Technology, China, in 2009. He is a Professor in the School of Information and Communication, Guilin University of Electronic Technology, China.
Lin Zheng received the M.E. degree in communication engineering from Guilin University of Electronic Technology, China, in 1996; and the Ph.D. degree in information and communication systems from Xidian University, China, in 2006. He is a Professor in the School of Information and Communication, Guilin University of Electronic Technology, China.

Hongbing Qiu received the M.E. degree in communication and electronic system from Xidian University, China, in 1991; and the Ph.D. degree in information and communication engineering from Xidian University, China, in 2004. He is a Professor in the School of Information and Communication, Guilin University of Electronic Technology, China.

Shi Shu received the B.S. degree in mathematics from Huaqiao University, China, 1983; and the Ph.D. degree in Computational Mathematics from Xiangtan University, China, 2004. He is a Professor in the School of Mathematical and Computational Science, Xiangtan University, China.