IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received July 30, 2016, accepted September 12, 2016, date of publication September 22, 2016, date of current version October 15, 2016.

Digital Object Identifier 10.1109/ACCESS.2016.2611619

Queueing Characteristics of the Best Effort Network Coding Strategy

XIANXU LI, QING CHANG, AND YONG XU
School of Electronic and Information Engineering, Beihang University, Beijing 100191, China
Corresponding author: Y. Xu. (xuyong1518@163.com)

This work was supported by the National Natural Science Foundation of China under Grant 61471021.

ABSTRACT Asynchronous network coding has the potential to improve wireless network performance compared with simple routing. However, to achieve the maximum network coding gain, the encoding node consumes a few computing and storage resources that may be unaffordable for wireless sensor networks such as CubeSats. An analogous threshold strategy, called best effort network coding (BENC), which requires only minimal storage resources and no computing resources, is investigated in this paper as an efficient and convenient method of network coding. In this strategy, a new packet arrival evicts the head packet when the queue is full to avoid excessively long waits. Moreover, in contrast to other methods that require a queue for each flow, the BENC uses only one queue for the two coded flows. In addition, the problem of time interval distribution for the output flow, which combines two independent flows, is investigated, and the network coding gain is then analyzed. While the maximum coding gain requires infinite buffer capacity under two independent Poisson arrivals with the same transmission rates, the calculation results show that the BENC needs only 4 buffers to achieve 90% of the maximum coding gain and can reach 99% of the maximum coding gain using 50 buffers. These results are verified by numerical simulations.

INDEX TERMS Network coding, best effort, queueing analysis, queue capacity, wireless sensor networks.

I. INTRODUCTION

Network coding is a new forwarding method that breaks the established pattern (which regards information flow as a commodity) by mixing several packets into one coded packet using algebraic operations. These coded packets can then deliver the contents of more than one packet in a single transmission and, thus, reduce transmission times through a router (or access point), increasing throughput and decreasing power consumption [1]. Based on the maximal flow-minimal cut theorem in graph theory [2], a sender and receiver cannot communicate at a rate greater than the maximal flow (or the minimal cut). In a butterfly network, traditional 'store–and–forward' routing cannot achieve maximal flow. However, in their seminal work, Ahlswede et al. [3] showed that network coding can theoretically achieve the maximal flow rate. Subsequently, network coding attracted a great deal of attention and studies proliferated rapidly. Research into network coding applications can be divided into intra-session [4] and inter-session [5] categories based on the source nodes from which the uncoded packets were generated. To increase the coding opportunities, these two schemes have been extended into different protocol layers using customized strategies. Physical network coding attempted to turn the "interference" into good use [6]. MAC scheduling schemes, which decide whether and how intermediate nodes encode packets, were considered in [7]–[9] to optimize the throughput performance. In the network layer, methodologies to optimize the system's performance under certain constraints were established taking both synchronous and asynchronous flows and infinite or finite buffers into account.

One important domain of application is in two-way relay networks with stochastic arrivals, in which a situation could occur where no packets are available for the other connection. The primary problem here is to design a mechanism to terminate excessively long waits, otherwise the system becomes unstable and the mean wait time grows unboundedly [10]. Several relevant aspects have already been explored in the literature.

One methodology involves terminating waits using a probability. Chieochan et al. [11] proposed a queueing model with a finite buffer at the relay that could be shared by two independent injected source packets. Whether packets were transmitted or received depended on a probability pertaining to each link in every time slot. Abdelrahman et al. [12] proposed a queuing model with infinite capacity for each flow. The packet waits for an exponential time-out period before forwarding a packet without coding. Ding et al. [13] considered a two-way relay network with Bernoulli asymmetric arrival processes with a finite capacity. The relay node forwards one uncoded packet with a probability relevant to the buffer length.

Some strategies have been designed to deal with the long wait issue. The COSE strategy was proposed in [14], in which Poisson arrival rates were assumed. A fixed time interval was configured to maintain a coding opportunity. Packets left uncoded would be released from the waiting queue and the process restarted. Paramanathan et al. [15] tried to guarantee each flow 50% of its buffer by dropping the oldest packets to manifest a high coding gain under a short-term asymmetric rate. They implemented this mechanism on their platform.

Several publications have pursued optimization problems that combine performance and cost objectives. Yuan et al. [16] provided an optimal trade-off between coding opportunities and packet-delay with n input flows, but this method required complex computing to calculate the probability matrix. The authors in [17] and [18] devised a mechanism that forwarded uncoded packets based on a cost threshold such as the energy consumption required for transmission by the relay. Several relevant policies such as cost threshold were considered, and the results of optimizing objectives relevant to delay or cost were presented. Ciftcioglu et al. [19] and Davri et al. [20] moved the buffers from the relay node to the source nodes to enhance the throughput of the MAC channel. Zohdy et al. [21] focused on efficiency gains and took delay-related performance into account as an optimization constraint rather than as an objective.

Few works have concentrated on the statistical characteristics of the output yielded by network coding. The output distribution of the time intervals for synchronous network coding scenarios was presented in [22]. The theoretical results showed that the distribution was an asymptotical exponent when the rates of the two independent exponential input flows were the same. Alsebae et al. [23] investigated the output distribution of an M/D/1 queueing model under a synchronous network coding scenario. In their study, only encoded packets were included in the output flow.

The contributions of the present paper are twofold. First, to achieve an efficient asynchronous network coding method, an analogous threshold queueing strategy with finite queuing capacity, called Best Effort Network Coding (BENC), is investigated. The BENC needs few storage resources and no computing resources to achieve network coding gains close to the theoretical maximum, and it is easily deployed on encoding nodes. Second, network coding output statistical characteristics, such as time interval distribution, mean waiting queue length and delay, are also investigated.

This study is organized as follows: the queuing model and the BENC strategy are presented in Section II. Queueing characteristics such as the time interval distribution of output for BENC and its expectation, as well as the mean waiting queue length are studied in Section III. Then, the performance is investigated using simulated independent exponential input processes with the same parameters in Section IV. Finally, conclusions are presented in Section V.

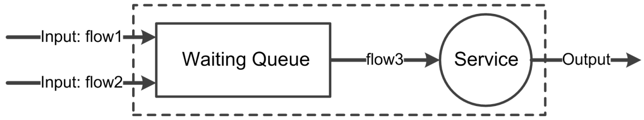


FIGURE 1. Queue model of an encoding node.

II. BEST EFFORT NETWORK CODING MODEL

Fig. 1 shows the queuing model of an encoding node, which consists of two independent data flows named flow 1 and flow 2, respectively, one queue model and one output flow. The data packet from each flow arrives after the time interval T whose distribution is exponential with λ_1 and λ_2, respectively. The packets arrive at the encoding node and are served to generate the output flow. The queue model in the encoding node is divided into two parts. The first part is the waiting queue, which has a finite buffer length, N, and the second is a service queue that stores and forwards packets. In the second part of this queuing model, the classical queueing theory of GI/GI/1/N is able to analyze the performance theoretically. The network coding occurs in the first part, for which there is not yet sufficient theoretical analysis.

Therefore, this study focuses on the first part in Fig. 1. Packets from flow 1 and flow 2 will be processed in the waiting queue using the BENC strategy to produce flow 3. Then, flow 3 becomes the input flow to the second part, where it is transmitted with the general service distribution to produce the output flow leaving the encoding node. Therefore, the prime task here is to obtain the probability distribution of flow 3.

A. THE BENC STRATEGY

In the BENC strategy, if the buffer is full or a matched packet arrives, then a packet should leave the waiting queue. The specific data flow diagram for BENC is shown in Fig. 2. The relay maintains two virtual, parallel, finite buffers; each virtual buffer accommodates packets received from each source. By "virtual" we mean that one real buffer is sufficient in a real system. When a new packet arrives, BENC checks whether the virtual buffer of the other flow is empty. If the other buffer is not empty, the newly arrived packet is encoded with a packet in the other flow and departs the waiting queue immediately. However, if the other buffer is empty when the new packet arrives, it will enter its own virtual buffer to wait for a future encoding opportunity. In addition, if its own buffer is full, the head packet leaves the waiting queue, opening one slot for the new arrival. Conversely, when a flow's virtual buffer is not full, packets already in the queue remain in the queue. Based on the BENC strategy, packets will not be lost from the waiting queue when the queue is full.

When neither of virtual queues are empty, the network coded packet will be generated and depart the queue immediately, therefore, there is practically only one queue which

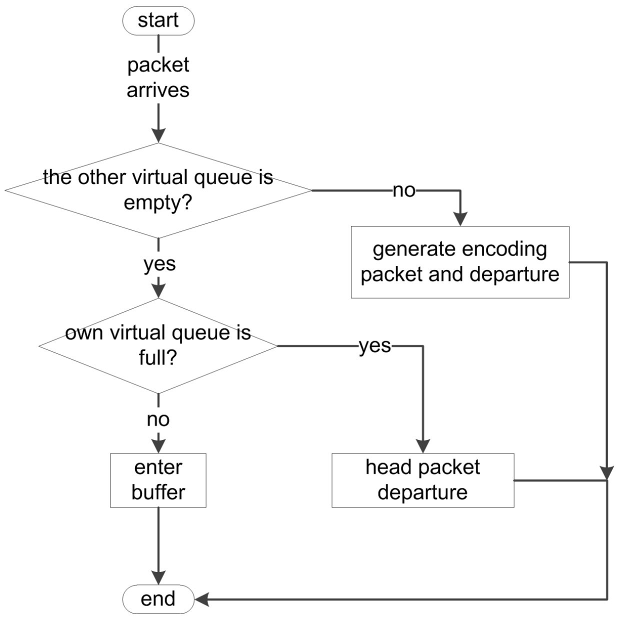


FIGURE 2. The best effort network coding (BENC) strategy.

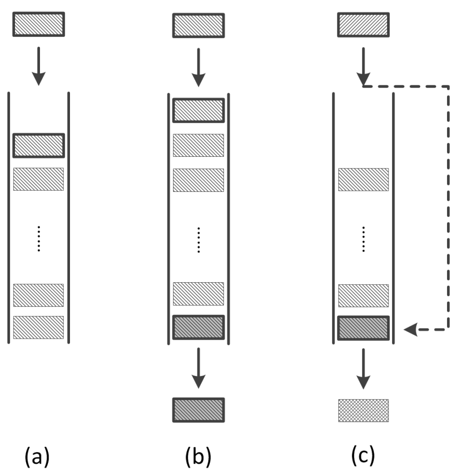


FIGURE 3. Illustration of the BENC strategy.

stores data of flow 1 or flow 2 at any given time. To clarify the scheme, Fig. 3 shows an example. The first part of the figure shows that a new arrival will enter the queue when the buffer is available. The second part shows that a new arrival will evict the head of the queue when the buffer is full. The third part shows that a new arrival of a different type will free one slot in the queue through network coding. Subsequently, the two encoded packets depart immediately.

B. THE BUFFER CAPACITY OF BENC

This study does not consider the time required to generate an encoding packet (which will depart the queue immediately) in the following analysis. Furthermore, only two independent input flows are considered. The capacity of both virtual buffers is N. The elements (k, l) represent the system state, which consists of the lengths of the virtual buffers. Here, k and l stand for the buffer lengths of flow 1 and flow 2, respectively. Based on the strategy described in Section II-A, the transition diagram is illustrated in Fig. 4. For the BENC strategy, we have $k^*l = 0$, which means at least one buffer is empty. To save storage resources, a real system requires only one buffer to accommodate packets from two input flows.

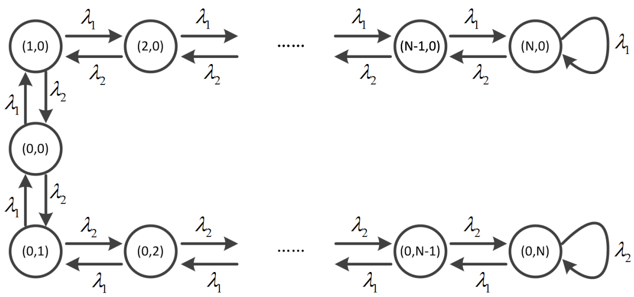


FIGURE 4. State transition diagram for the Markov chain.

III. PROBLEM FORMULATION

The probability distribution for the inter-arrival time of input flows is exponential with λ_1 and λ_2, respectively. The random variable $\hat{T}$ represents the time interval of flow 3.

A. STEADY STATE PROBABILITY

The column for the steady state probability, $\mathbf{W}$, is defined as:

$$W = \left[P\,\omega_N^1, \ldots, P\,\omega_1^1, P\,\omega_0, P\,\omega_1^1, \ldots, P\,\omega_N^2\right]^T$$

Although the capacity of the relay buffer is N, the state of this system is $2N + 1$. Let ω_i^1, ω_i^2 and ω_0 denote the state $(i,\ 0)$, $(0,\ i)$, and $(0,\ 0)$, respectively, where $i = 1, 2, \ldots, N$.

Ω^1 and Ω^2 represent sets of ω_i^1 and ω_i^2. Hence, $\Omega = \Omega^1, \omega_0, \Omega^2$ is the state set of this model. The steady state probability, $\mathbf{W}$, can be calculated as follows:

$$\mathbf{W}^T \boldsymbol{\Lambda} = \mathbf{W}^T, \tag{1}$$

where $\boldsymbol{\Lambda}$ is the one-step transition probability matrix as shown at the top of the next page.

The result of the steady state probability is expressed as:

$$P\left\{\omega_i^j\right\} = \frac{\lambda_j^{N+i}\lambda_{3-j}^{N-i}}{\sum\limits_{k=0}^{2N}\lambda_1^{2N-k}\lambda_2^k}, \quad j = 1, 2 \tag{2}$$

$$P\left\{\omega_0\right\} = \frac{\lambda_1^N\lambda_2^N}{\sum\limits_{k=0}^{2N}\lambda_1^{2N-k}\lambda_2^k}, \tag{3}$$

where i ranges from 1 to N.

B. THE EXPECTATION OF TIME INTERVAL

The distribution of the time interval for flow 3 can be written as:

$$P\left\{\hat{T} > t\right\} = \sum_{\omega\in\Omega} P\left\{\hat{T} > t \,\middle|\, \omega\right\} P\left\{\omega\right\}. \tag{4}$$

$$\mathbf{\Lambda} = \begin{bmatrix} \frac{\lambda_1}{\lambda_1+\lambda_2} & \frac{\lambda_2}{\lambda_1+\lambda_2} & & & & \\ \frac{\lambda_1}{\lambda_1+\lambda_2} & 0 & \frac{\lambda_2}{\lambda_1+\lambda_2} & & 0 & \\ & \frac{\lambda_1}{\lambda_1+\lambda_2} & 0 & \frac{\lambda_2}{\lambda_1+\lambda_2} & & \\ & & \ddots & \ddots & \ddots & \\ & 0 & & \frac{\lambda_1}{\lambda_1+\lambda_2} & 0 & \frac{\lambda_2}{\lambda_1+\lambda_2} \\ & & & & \frac{\lambda_1}{\lambda_1+\lambda_2} & \frac{\lambda_2}{\lambda_1+\lambda_2} \end{bmatrix}$$

The conditional probability where the time interval $\hat{T}$ is greater than t can be calculated as follows:

$$P\left\{\hat{T} > t \middle| \omega_i^j\right\} = e^{-(\lambda_1+\lambda_2)t} \sum_{k=0}^{N-i} \frac{(\lambda_j t)^k}{k!}, \quad j = 1, 2 \tag{5}$$

$$P\left\{\hat{T} > t \middle| \omega_0\right\} = e^{-(\lambda_1+\lambda_2)t} \left(\sum_{k=0}^{N} \frac{(\lambda_1 t)^k + (\lambda_2 t)^k}{k!} - 1 \right), \tag{6}$$

where i ranges from 1 to N.

The conditional probability can be expressed in vector form as follows:

$$\mathbf{\Phi} = \begin{bmatrix} P\left\{\hat{T} > t \middle| \omega_N^1\right\}, \ldots, P\left\{\hat{T} > t \middle| \omega_1^1\right\}, P\left\{\hat{T} > t \middle| \omega_0\right\}, \\ P\left\{\hat{T} > t \middle| \omega_1^2\right\}, \ldots, P\left\{\hat{T} > t \middle| \omega_N^2\right\} \end{bmatrix}^T. \tag{7}$$

Then, Eqn. (4) can be simplified to

$$P\left\{\hat{T} > t\right\} = \sum_{\omega \in \Omega} P\left\{\hat{T} > t \middle| \omega\right\} P\{\omega\} = \mathbf{W}^T \mathbf{\Phi}. \tag{8}$$

The probability distribution of the random variable $\hat{T}$ is denoted by $\mathrm{F}_{\hat{T}}(t)$, which is defined as follows:

$$\mathrm{F}_{\hat{T}}(t) = P\left\{\hat{T} < t\right\} = 1 - \mathbf{W}^T \mathbf{\Phi}. \tag{9}$$

Then the expectation of the random variable $\hat{T}$ can be obtained as shown below:

$$\begin{aligned} \mathrm{E}\left[\hat{T}\right] &= \int_{t=0}^{\infty} t \cdot \frac{d}{dt}\left[1 - \mathbf{W}^T \mathbf{\Phi}\right] dt \\ &= \mathbf{W}^T \cdot \int_{t=0}^{\infty} t \cdot \left(-\frac{d}{dt}\mathbf{\Phi}\right) dt \\ &= \frac{\lambda_1 - \lambda_2}{\lambda_1^{2N+1} - \lambda_2^{2N+1}} \\ &\cdot \left\{ \begin{aligned} &\frac{\lambda_1^{N-1}\lambda_2^{2N}}{(\lambda_1+\lambda_2)^N} + \lambda_2^{N+1}\frac{\lambda_1^{N-1}-\lambda_2^{N-1}}{\lambda_1-\lambda_2} \\ &+\lambda_1^{N-1}\lambda_2^{N-1}\frac{(\lambda_1^2+\lambda_2^2+\lambda_1\lambda_2)(\lambda_1+\lambda_2)^N - \lambda_1^{N+2} - \lambda_2^{N+2}}{(\lambda_1+\lambda_2)^{N+1}} \\ &+\frac{\lambda_2^{N-1}\lambda_1^{2N}}{(\lambda_1+\lambda_2)^N} + \lambda_1^{N+1}\frac{\lambda_1^{N-1}-\lambda_2^{N-1}}{\lambda_1-\lambda_2} \end{aligned} \right\}. \end{aligned} \tag{10}$$

C. THE EXPECTATION OF NETWORK CODING TIME INTERVAL

The output of flow 3 consists of both network coded packets and uncoded packets. The time interval between network coded packets is defined as T_{NC}. The cardinality of the state set Ω is $2N + 1$ when the buffer length is $N$for each flow. The buffer length state changes when a new arrival packet is processed. It is only when the state is ω_N^1 or ω_N^2 that departing packets could be uncoded. Otherwise, departing packets will always be encoded. The conditional probability that T_{NC} is greater than t is calculated as follows:

$$P\left\{T_{NC} > t | \omega_i^j\right\} = e^{-\lambda_{3-j}t}, \quad j = 1, 2 \tag{11}$$

$$P\{T_{NC} > t|\omega_0\} = e^{-\lambda_1 t} + e^{-\lambda_2 t} - e^{-\lambda_1 t}e^{-\lambda_2 t}, \tag{12}$$

where i ranges from 1 to $N - 1$.

The probability that T_{NC} is greater than t is the sum of the conditional probability multiplied by the steady state probability, as expressed in Eqn. (13).

$$\begin{aligned} P\{T_{NC} > t\} = &\sum_{i=1}^{N-1} \begin{pmatrix} P\left\{T_{NC} > t|\omega_i^1\right\} P_{NC}\left\{\omega_i^1\right\} \\ + P\left\{T_{NC} > t|\omega_i^2\right\} P_{NC}\left\{\omega_i^2\right\} \end{pmatrix} \\ &+ P\{T_{NC} > t|\omega_0\} P_{NC}\{\omega_0\}. \end{aligned} \tag{13}$$

However, the steady state probability (P_{NC}) is no longer the system steady state probability presented in Eqns. (2 and 3). P_{NC} can be easily calculated by steady state probability using the following equations:

$$P_{NC}\left\{\omega_i^j\right\} = \frac{P\left\{\omega_i^j\right\}}{\sum_{k=1}^{N-1}\left(P\left\{\omega_k^1\right\} + P\left\{\omega_k^2\right\}\right) + P\{\omega_0\}}, \quad j = 1, 2 \tag{14}$$

$$P_{NC}\{\omega_0\} = \frac{P\{\omega_0\}}{\sum_{k=1}^{N-1}\left(P\left\{\omega_k^1\right\} + P\left\{\omega_k^2\right\}\right) + P\{\omega_0\}}, \tag{15}$$

where i ranges from 1 to $N - 1$.

The conditional probability and the steady state probability can be expressed in vector form as follows:

$$\mathbf{W_{NC}} = \begin{bmatrix} P_{NC}\left\{\omega_{N-1}^1\right\}, \cdots, P_{NC}\left\{\omega_1^1\right\}, P_{NC}\{\omega_0\}, \\ P_{NC}\left\{\omega_1^2\right\}, \cdots, P_{NC}\left\{\omega_{N-1}^2\right\} \end{bmatrix}^T,$$

$$\mathbf{G_{NC}} = \begin{bmatrix} e^{-\lambda_2 t}, \cdots, e^{-\lambda_2 t}, e^{-\lambda_1 t} + e^{-\lambda_2 t} - e^{-\lambda_1 t}e^{-\lambda_2 t}, \\ e^{-\lambda_1 t}, \cdots, e^{-\lambda_1 t} \end{bmatrix}^T.$$

Then, Eqn. (13) can be simplified to

$$P\{T_{NC} > t\} = \mathbf{W}_{\mathbf{NC}}^{T}\mathbf{G}_{\mathbf{NC}}, \tag{16}$$

and the expectation of the random variable T_{NC}can be obtained as follows:

$$\begin{aligned}
&\mathrm{E}[T_{NC}] \\
&= \int_{t=0}^{\infty} t \cdot \frac{d}{dt}\left[1 - \mathbf{W}_{\mathbf{NC}}^{T}\mathbf{G}_{\mathbf{NC}}\right]dt \\
&= \mathbf{W}_{\mathbf{NC}}^{T} \cdot \int_{t=0}^{\infty} t \cdot \left(-\frac{d}{dt}\mathbf{G}_{\mathbf{NC}}\right)dt \\
&= \frac{-\lambda_1^{2N+1} - \lambda_1^{2N}\lambda_2 + \lambda_1^{N+1}\lambda_2^{N} + \lambda_1\lambda_2^{2N} - \lambda_1^{N}\lambda_2^{N+1} + \lambda_2^{2N+1}}{(\lambda_1+\lambda_2)\left(-\lambda_1^{2N}\lambda_2 + \lambda_1\lambda_2^{2N}\right)} \\
&= \frac{(\lambda_1+\lambda_2)\sum\limits_{i=0}^{2N-1}\lambda_1^{2N-1-i}\lambda_2^{i} - \lambda_1^{N}\lambda_2^{N}}{(\lambda_1+\lambda_2)\lambda_1\lambda_2\sum\limits_{i=0}^{2N-2}\lambda_1^{2N-2-i}\lambda_2^{i}}.
\end{aligned} \tag{17}$$

Assuming λ_2 if greater than λ_1, according to Eqn. (17), with the increase of λ_2 or buffer capacity N, the expectation of time interval of encoded packets tends toward $1/\lambda_1$, due to the rate of encoded fraction of total output flow is determined by the rate of slower input flow, which is λ_1.

D. MEAN WAITING QUEUE LENGTH

The mean waiting queue lengths, L_Q^1 and L_Q^2, for flows 1 and 2, respectively, can be obtained by using the system steady state probability as follows:

$$\begin{aligned}
L_Q^j &= 0 \times P\{\omega_0\} + \sum_{i=1}^{N} i \times P\left\{\omega_i^j\right\} \\
&= \frac{\lambda_j^{N+1}\left[N\lambda_j^{N+1} - (N+1)\lambda_j^{N}\lambda_{3-j} + \lambda_{3-j}^{N+1}\right]}{(\lambda_1-\lambda_2)\left(\lambda_1^{2N+1} - \lambda_2^{2N+1}\right)}, \\
&\qquad j = 1, 2. \\
&= \lambda_j^{N+1}\frac{\sum\limits_{k=0}^{N-1}\lambda_j^{N-1-k}\sum\limits_{i=0}^{k}\lambda_j^{k-i}\lambda_{3-j}^{i}}{\sum\limits_{k=0}^{2N}\lambda_1^{2N-k}\lambda_2^{k}}
\end{aligned} \tag{18}$$

The mean waiting queue length varies with the rate of input flow and the capacity of queue. Assuming the capacity of queue is fixed, according to Eqn. (18), the queue length of flow j decreases with the increase of rate of flow $3-j$, when the rate of the flow $3-j$ is higher than the rate of flow j, where j has a value of 1 or 2. The reason is as follows. The packets from flow $3-j$ arrives faster than the packets from flow j, therefore, the packets from flow j stay in the waiting queue for a short time, and then encoded packets are generated and depart the queue, as a result, the queue length of flow j decreases. On the contrary, the queue length of flow $3-j$ increases.

E. MEAN PACKET WAIT TIME

Little's Law holds if a queueing process is regenerative [24]. The BENC process is regenerative, thus, the mean wait time, W_Q, for each flow can be calculated by $L_Q^j = \lambda_j W_Q^j$, where j is 1, 2.

$$W_Q^j = \lambda_j^{N}\frac{\sum\limits_{k=0}^{N-1}\lambda_j^{N-1-k}\sum\limits_{i=0}^{k}\lambda_j^{k-i}\lambda_{3-j}^{i}}{\sum\limits_{k=0}^{2N}\lambda_1^{2N-k}\lambda_2^{k}}, \quad j = 1, 2. \tag{19}$$

IV. PERFORMANCE ANALYSIS AND SIMULATION

A. THE NETWORK CODING GAIN

The expectation rate, $\hat{\lambda}$, of flow 3 is $\hat{\lambda} = 1/\left(\mathrm{E}\left[\hat{T}\right]\right)$. The result is presented in Eqn. (20) when λ_1 and λ_2 are equal to λ in Eqn. (10):

$$\hat{\lambda} = \lambda\frac{2N+1}{\frac{1}{2^N} + 2N - \frac{1}{2}}. \tag{20}$$

1) WHEN THE BUFFER IS ABSENT

The rate of flow 3 is $\hat{\lambda} = 2\lambda$ when N is 0. Because λ_1 equals λ_2, the rate of flow 3 is equal to $\lambda_1 + \lambda_2$. In this case, the process of flow 3 is the sum of the two independent exponential distributed processes with parameters λ_1 and λ_2. The result of $\hat{\lambda}$ can then be verified.

2) WHEN THE CAPACITY OF THE BUFFER IS INFINITE

The asynchronous network coding process of flow 3 will degenerate to the synchronous network coding. The rate of the asynchronous network coding asymptotically tends toward λ as the capacity of the buffer increases to infinity, which can be calculated as follows:

$$\hat{\lambda} = \lim_{N\to\infty} \lambda\frac{2N+1}{\frac{1}{2^N} + 2N - \frac{1}{2}} = \lambda. \tag{21}$$

In this case, the encoding node reaches the maximum coding gain, which is 2. Meanwhile, the result also implies that the output flow rate of a synchronous network coding is λ when the two independent Poisson processes have the same rate, λ. The same result is shown in [14] as well.

3) WHEN THE CAPACITY OF BUFFER IS N

From the preceding analysis, the maximum network coding gain is 2 when two independent processes have the same rate. The BENC strategy will achieve the maximum gain when the buffer capacity is infinite. However, in real networks, this theoretical maximum is obviously unreasonable. A better question is: How large a buffer is required to achieve 99% of the maximum coding gain? The answer is shown below.

The network coding gain is the ratio of the number of forwarding transmissions required to the number of transmissions required by the network coding strategy to deliver the packets with the same size. The output of flow 3 consists of both encoded and uncoded packets. The output rate of flow 3 is the sum of the network coding rate, λ_{NC}, and the uncoded flow rate. The network coding gain can then be calculated by $(2\lambda_{NC} + \hat{\lambda} - \lambda_{NC})/\hat{\lambda} = 1 + \lambda_{NC}/\hat{\lambda}$. The expectation rate is defined as $\lambda_{NC} = 1/(E[T_{NC}])$. The result is presented in Eqn. (22) when the parameters λ_1 and λ_2 are equal to λ in Eqn. (17):

$$\lambda_{NC} = \frac{4N-2}{4N-1}\lambda. \tag{22}$$

The buffer size N is the smallest integer that is no less than the result from solving Eqn. (23) to achieve x percent of the maximum coding gain. Therefore, buffer sizes of 50, 10 and 4 can reach 99%, 95% and 90% of the maximum coding gain, respectively.

$$1 + \frac{(4N-2)\left(\frac{1}{2^N} + 2N - \frac{1}{2}\right)}{(4N-1)(2N+1)} = 2 \times x\%. \tag{23}$$

B. SIMULATION

The theoretical results discussed in Section 3 such as the expectation time interval of flow 3, the waiting queue length for each flow and the network coding gain are investigated in this section. In particular, the situation when λ_1 equals λ_2 is analyzed here. The distribution of these two input flows is exponential. Without loss of generality, $\lambda_1 = \lambda_2 = \lambda = 1$ is assumed, which can be easily achieved by properly choosing a time scale. The duration of every simulation is at minimum 100,000 time units.

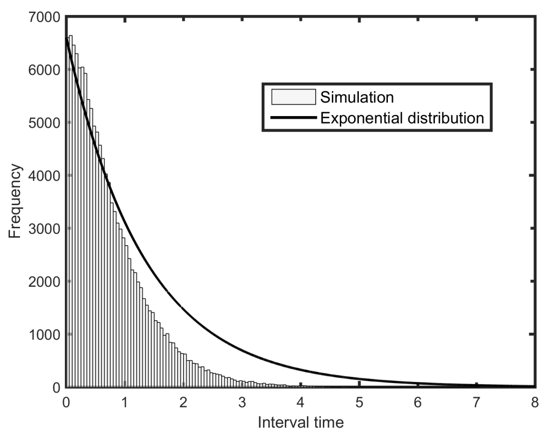


FIGURE 5. Time interval distribution with $N = 1$.

1) THE TIME INTERVAL DISTRIBUTION OF FLOW 3

The time interval distribution of flow 3 depends on λ_1, λ_2 and the waiting queue length N in Eqn. (10). In addition, the hazard function of the time interval for flow 3 is no longer a constant; therefore, flow 3 is no longer a Poisson process [24]. The exponential distribution curves in Fig. 5, Fig. 6, and Fig. 7 are obtained by curve fitting method based on the simulation data. Fig. 5 shows that when the queue capacity is 1, the time interval frequency histogram does not match the exponential distribution. However, as the capacity of waiting queue increases, the distribution is asymptotically exponential, as shown in Fig. 6 and Fig. 7. When the capacity is greater than 4, the distribution is almost exponential.

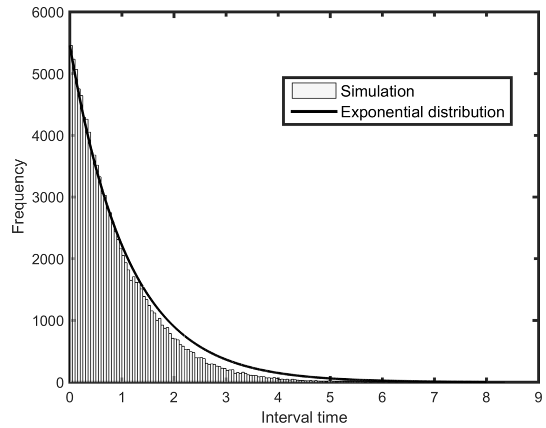


FIGURE 6. Time interval distribution with $N = 4$.

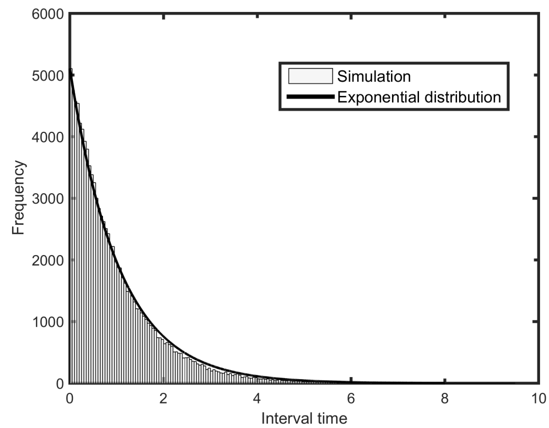


FIGURE 7. Time interval distribution with $N = 10$.

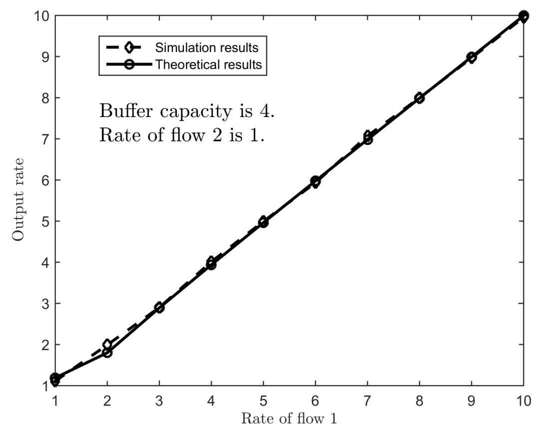


FIGURE 8. The output flow rate with two asymmetric flow rates.

Fig. 8 shows the output flow rate with asymmetric source rates. The rate ratio of the two flows ranges from 1 to 10. The output rate is almost equal to the rate of the fastest flow, which means that the packets from the slower flow are nearly entirely encoded.

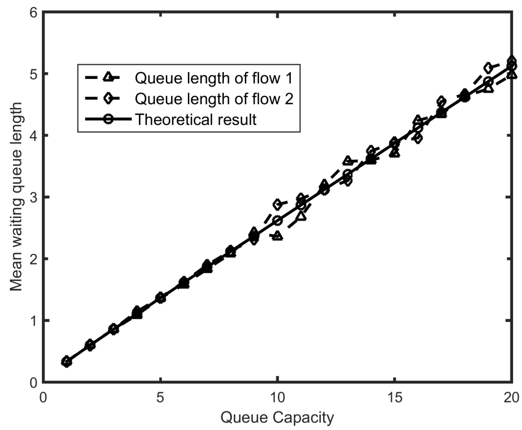


FIGURE 9. Mean waiting queue lengths.

2) MEAN WAITING QUEUE LENGTHS OF FLOW 1 AND FLOW 2

When $\lambda_1 = \lambda_2 = \lambda$, Eqn. (18), which calculates the mean waiting queue lengths of flow 1 and flow 2, can be simplified to $L_Q^1 = L_Q^2 = N(N+1)/2(2N+1)$. The theoretical result shows that the mean waiting queue lengths of flow 1 and flow 2 are equal and are related only to the queue capacity. Fig. 9 shows the mean waiting queue length plotted as a function of the queue capacity. The solid theoretical line lies between the two simulation lines; however, the simulation results are extremely close to the theoretical result.

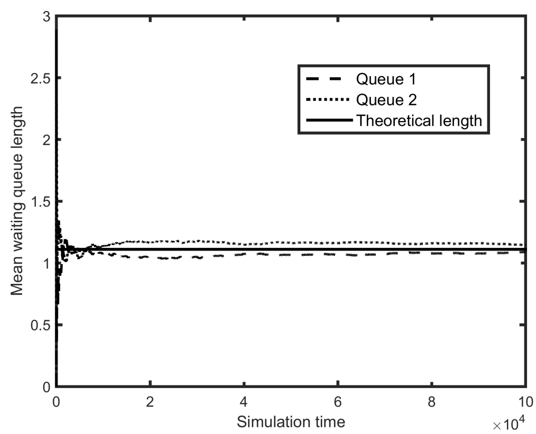


FIGURE 10. Mean waiting queue length with $N = 4$.

Fig. 10 and Fig. 11 show the mean waiting queue lengths plotted as functions of the simulation time. The queue capacities in Fig. 10 and Fig. 11 are 4 and 10, respectively. The solid lines present the theoretical results. The dashed lines show the results for queue 1, while the dotted lines show the results for queue 2. The theoretical mean waiting queue lengths are $L_Q^1 = L_Q^2 = 4(4+1)/2(2\times 4+1) \approx 1.1111$ and $L_Q^1 = L_Q^2 = 10(10+1)/2(2\times 10+1) \approx 2.6190$ when the queue capacities are 4 and 10, respectively. The simulation results show that the mean queue length oscillates between 0 and the queue capacity at the beginning, but tends to stabilize quickly. Hence, the system will be stable.

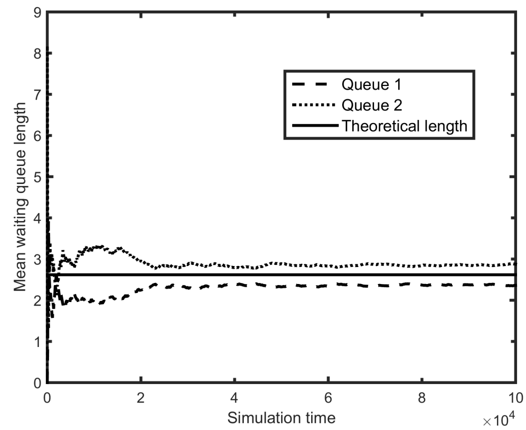


FIGURE 11. Mean waiting queue length with $N = 10$.

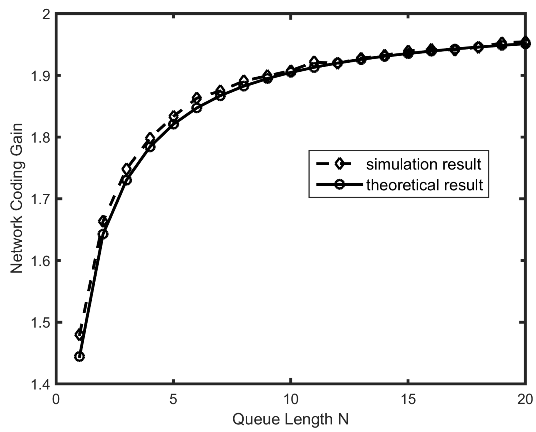


FIGURE 12. Simulation of network coding gain.

3) NETWORK CODING GAIN

The network coding gain is illustrated in Fig. 12. The diamond and circle lines respectively represent the simulated and theoretical results of the network coding gain. The simulation results show that the network coding gains are 1.798 and 1.908 when N equals 4 and 10, respectively. This simulation verifies the theoretical result for network coding gain. That is to say, if 90% of the maximum coding gain is sufficient for a communication system, the length of the waiting queue for each flow needs to be only 4. The output flow rate, $\hat{\lambda}$, is 1.19λ when the network coding gain is 1.8.

The mean result of the network coding gain from simulation is consistent with the theoretical analysis. Therefore, the BENC strategy is an efficient and succinct way to obtain sufficient network coding gain at the encoding node.

V. CONCLUSION

In this study, the queueing characteristics of network coding have been investigated. A novel strategy called Best Effort Network Coding (BENC) was designed and formulated using a Markov chain. The problem of the distribution of the time interval for the output flow has been solved. In addition, the expectations were presented in analytical form, including the time interval for the output flow, the network coding time interval and the mean waiting queue length. Moreover, the performance of this strategy in one particular case has been analyzed and validated through simulation. As shown by the simulation, the waiting queue system is stable using the BENC strategy. The distribution of the output flow is asymptotically exponential as the queue capacity increases. Furthermore, network coding gain is achieved efficiently by the BENC strategy without requiring complex encoding calculations at the encoding node.

REFERENCES

[1] T. Ho and D. Lun, *Network Coding: An Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 2008, pp. 1–5.

[2] W. K. Chen, "Maximum Flow in Networks," in *Theory of Nets: Flows in Networks*. England, U.K.: Cambridge Univ. Press, 1990.

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

[4] S. Chachulski, M. Jennings, S. Katti, D. Katabi, "Trading structure for randomness in wireless opportunistic routing," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 4, pp. 169–180, Oct. 2007.

[5] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Médard, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *ACM SIGCOMM Comput. Commun. Rev.*, vol. 36, no. 4, pp. 243–254, Oct. 2006.

[6] S. Zhang, S. C. Liew, and P. P. Lam, "Hot topic: Physical-layer network coding," in *Proc. 12th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2006, pp. 358–365.

[7] P. Chaporkar and A. Proutiere, "Adaptive network coding and scheduling for maximizing throughput in wireless networks," in *Proc. 13th Annu. ACM Int. Conf. Mobile Comput. Netw. (MobiCom)*, 2007, pp. 135–146.

[8] Y. E. Sagduyu and A. Ephremides, "On joint mac and network coding in wireless ad hoc networks," *IEEE Trans. Inf. Theory*, vol. 53, no. 10, pp. 3697–3713, Oct. 2007.

[9] Y. P. Hsu, N. Abedini, N. Gautam, A. Sprintson, and S. Shakkottai, "Opportunities for network coding: To wait or not to wait," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, Jul./Aug. 2011, pp. 791–795.

[10] X. He and A. Yener, "On the energy-delay trade-off of a two-way relay network," in *Proc. 42nd Annu. Conf. Inf. Sci. Syst. (CISS)*, Princeton, NJ, USA, Mar. 2008, pp. 865–870.

[11] S. Chieochan, S. Hossain, T. Issariyakul, and D. Niyato, "Opportunistic network coding and dynamic buffer allocation in a wireless butterfly network," in *Proc. Global Telecommun. Conf. (GLOBECOM)*, Honolulu, HI, USA, Nov./Dec. 2009, pp. 1–6.

[12] O. H. Abdelrahman and E. Gelenbe, "Performance trade-offs in a network coding router," in *Proc. 19th Int. Conf. Comput. Commun. Netw. (ICCCN)*, Zurich, Switzerland, Aug. 2010, pp. 1–6.

[13] L. H. Ding, F. Yang, L. Qian, L. Liu, and P. Wu, "Cost-delay trade off of network coding in asymmetric two-way relay networks," in *Proc. Int. Symp. Netw. Coding*, Beijing, China, Jul. 2011, pp. 1–4.

[14] Y. Ma, W. Li, P. Fan, K. B. Letaief, and X. Liu, "On the Characteristics of Queueing and Scheduling at Encoding Nodes for Network Coding," *Int. J. Commun. Syst.*, vol. 22, no. 6, pp. 755–772, Jun. 2009.

[15] A. Paramanathan, D. E. Lucani, and F. H. P. Fitzek, "On the need for buffer optimization strategies for inter-session network coding," in *Proc. IEEE Int. Conf. Commun. Workshops (ICC)*, Sydney, NSW, Australia, Jun. 2014, pp. 343–348.

[16] Y. Yuan, K. Wu, W. Jia, and Y. Peng, "On the queueing behavior of inter-flow asynchronous network coding," *Comput. Commun.*, vol. 35, no. 13, pp. 1535–1548, Jul. 2012.

[17] Y. P. Hsu, N. Abedini, N. Gautam, A. Sprintson, and S. Shakkottai, "Opportunities for network coding: To wait or not to wait," in *Proc. IEEE Int. Symp. Inf. Theory (ISIT)*, St. Petersburg, Russia, Jul./Aug. 2011, pp. 791–795.

[18] A. Mohapatra, G. Natarajan, S. Srinivas, and S. Alex, "Network coding decisions for wireless transmissions with delay consideration," *IEEE Trans. Commun.*, vol. 62, no. 8, pp. 2965–2976, Aug. 2014.

[19] E. N. Ciftcioglu, Y. E. Sagduyu, R. A. Berry, and A. Yener, "Cost-delay tradeoffs for two-way relay networks," *IEEE Trans. Wireless Commun.*, vol. 10, no. 12, pp. 4100–4109, Dec. 2011.

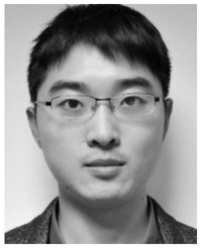
[20] E. C. Davri, K. Kontovasilis, and C. Skianis, "Optimized network coding efficiency under QoS constraints in two-way relay networks with time-outs," in *Proc. IEEE Int. Conf. Commun. (ICC)*, London, U.K., Jun. 2015, pp. 6157–6162.

[21] M. Zohdy, T. Elbatt, and M. Nafie, "Maximum throughput opportunistic network coding in two-way relay networks," in *Proc. IEEE Int. Conf. Commun. Workshop (ICCW)*, London, U.K., Jun. 2015, pp. 913–918.

[22] Y. Ma, W. Li, P. Fan, and X. Liu, "Queuing model and delay analysis on network coding," in *Proc. IEEE Int. Symp. Commun. Inf. Technol. (ISCIT)*, Beijing, China, Oct. 2005, pp. 112–115.

[23] A. Alsebae, M. S. Leeson, and R. Green, "Performance of a network coding queuing model with deterministic service," in *Proc. 9th Int. Symp. Commun. Syst., Netw. Digit. Signal Process. (CSNDSP)*, Manchester, Jul. 2014, pp. 988–993.

[24] S. M. Ross, "The exponential distribution and the poisson process," in *Introduction to Probability Models*, 11th ed. San Diego CA, USA: Academic, 2014.



XIANXU LI received the B.S. degree in communication and information system from the Beijing University of Posts and Telecommunications, Beijing, China, in 2010. He is currently pursuing the Ph.D. degree in communication and information system with Beihang University, Beijing. His research interest includes satellite navigation, satellite networks with crosslinks, and the applications of network coding.

QING CHANG received the B.S. degree in mathematics from the Beijing Normal University in 1982, and the M.S. degree in mathematics and the Ph.D. degree in communication and information system from Beihang University, in 1991 and 1995, respectively. Since 2000, he has been a Professor with the School of Electronic and Information Engineering, Beihang University. He is the Dean of Teaching-Research Section of Communication and Information System in his school. He is the author of one book, over 30 articles, and over ten patents. His research interests include satellite communication and navigation, and wireless networks.

YONG XU received the B.S. degree in communication from Ji Lin University in 2005, and the Ph.D. degree in electrical engineering from the Beihang University, China, in 2012. He has been a Post-Doctoral Fellow with the Department of Electronic Engineering, Beihang University in 2012. His professional research fields include communication and navigation theory and technology.