Modelling and Analysis of ARQ Mechanisms for Wireless Multi-hop Relay System

Soo-Yong Jeon* and Dong-Ho Cho†
School of Electrical Engineering and Computer Science
Korea Advanced Institute of Science and Technology (KAIST)
373-1 Guseong-dong Yuseong-gu Daejeon, Republic of Korea
TEL: +82-42-869-8067, FAX: +82-42-869-4042
Email: *syjeon@comis.kaist.ac.kr and †dhcho@ee.kaist.ac.kr

Abstract—Multi-hop relaying has attracted considerable attention recently as a key technique for improving cell coverage and throughput in wireless communication. We modelled various ARQ mechanisms that have been proposed for wireless multi-hop relay systems based on a Discrete Time Markov Chain (DTMC) technique. We analyzed the performance of the ARQ mechanisms with respect to delay and throughput. We also validated the proposed ARQ model and the performance analysis by comparing the analysis with simulation results.

I. INTRODUCTION

Recently, multi-hop relaying for wireless communication has attracted considerable attention in both industry and academia, in such standards as 3GPP, IEEE 802.16j and IEEE 802.11s. If multi-hop relaying is used, 3C problems (coverage, capacity and cost) in wireless communication systems can be solved easily. Fig. 1 shows a relay node (RN) deployment concept in a wireless multi-hop relay system. If the RN is located at the edge of a cell or in shadowing areas, the cell coverage is extended and the cell capacity increases with low cost, because the RN can use more transmission power and the channel quality of a mobile station (MS) increases due to low path loss. In addition, the power of MSs is conserved, due to the shortened distance across which it has to communicate. However, many problems must be solved if the relaying technique is to be used successfully. Additional radio resources are needed for a relaying channel, and the complexity of system increases due to additional interfaces with the RN. Further, the reliability and the quality of service (QoS) worsen, because the data is transmitted through many wireless links. Hence, in multi-hop relay systems, algorithms of greater complexity than in single-hop systems are required in order to guarantee reliability and QoS.

Many ARQ mechanisms have proposed to improve the reliability of multi-hop relaying systems [1]-[4]. The proposed ARQ mechanisms can be classified as end-to-end ARQ, hop-by-hop ARQ, and hybrid ARQ, according to the way in which they recover from errors. Each type of ARQ mechanism has its own benefits and drawbacks, which we will describe in a later section. In [4]-[7], the various ARQ mechanisms were modelled and their performances were compared. In [4][5], the performances of various ARQ mechanisms were analyzed in the single-packet transmission case. The multiple-packet transmission case was studied in [6]. In [7], an environment in which packets arrived continuously was studied, but channel capacity, queue size, and ARQ window size were not considered. In this paper, we will model the various ARQ mechanisms while taking into account continuous packet arrival, channel capacity, queue size, and ARQ window size. We will validate the proposed model by comparing its predictions with simulation results.

The rest of this paper is organized as follows. Section II introduces various ARQ mechanisms that have been proposed for multi-hop relay systems. Section III describes the proposed modelling procedure and evaluates the performance of the various ARQ mechanisms with respect to delay and throughput. In section IV, the proposed model is validated by comparing its predictions with simulation results. Section V concludes the paper.

II. ARQ MECHANISMS FOR MULTI-HOP RELAY SYSTEM

Figs. 2. shows the behavior of an end-to-end ARQ mechanism. With the end-to-end ARQ mechanism, the RN just relays the data and feedback between the source node and the destination node and takes no additional action. This ARQ mechanism uses a single error recovery protocol that covers a complete multi-hop route. So, the same protocol state is used for all hops and any link failure is managed by the source and the destination nodes without the aid of the RN. The end-to-end ARQ mechanism is very simple and deals with handover easily, because the the source node knows the status of transmitted ARQ blocks. However, it has many drawbacks, such as low transmission efficiency and long transmission
delay, because the source node retransmits the data that the RN has already received successfully but the destination node has failed to receive. In addition, the delay in the RN’s receiving the retransmitted data from the source node is very long, because it is always triggered by a retransmission timeout or a feedback message sent by the destination node.

Figs. 3. shows the behavior of a hop-by-hop ARQ mechanism. With this mechanism, the RN not only relays the data but also generates its own feedback. This ARQ mechanism uses an error recovery protocol that operates independently in each link. It has high transmission efficiency and short transmission delay, because any transmission failure can be recovered in each hop. However, the complexity of the RN increases and the data may be queued or lacked in the RN, because the data transmission of one link does not consider the data transmission of another link. In addition, when the MS performs a handover, it must report additional information about its block status to the BS, because the ARQ block status in the BS and the MS are managed separately.

As a hybrid ARQ approach, a relay ARQ mechanism was suggested in [2]-[4]. Figs. 4. shows the behavior of the relay ARQ mechanism. This ARQ mechanism uses a single error recovery protocol that covers the complete multi-hop route, just as does the end-to-end ARQ mechanism. However, in this case, the RN relays the data and feedback between the source and destination nodes and generates its own feedback and transmits it to the source node, as does the hop-by-hop ARQ mechanism. If the RN receives the data from the source successfully but fails to send the data to the destination node, the destination node sends an ARQ feedback message to communicate the NACK to the RN. The RN then relays it to the source node, together with an RN feedback message (RS_ACK) that tells the source node that the RN has received the packet successfully. The source node is thus informed by the RS_ACK and the NACK that the RN has received the data successfully but has failed to send the data to the destination node. Hence, the source node does not retransmit the data. It waits until the RN sends the data to the destination node successfully. If the RN fails to receive the data from the source node, it sends an RN feedback message that contains the RS_NACK, which indicates to the source node that the RN has failed to receive the packet. Upon receiving the RS_NACK, the source node recognizes that the RN fails to receive the data and retransmits it. This relay ARQ mechanism has high transmission efficiency and short transmission delay, roughly equivalent to those of the hop-by-hop ARQ mechanism. In addition, the relay ARQ mechanism deals with MS handover easily, just as does the end-to-end ARQ mechanism, because the source node knows the status of transmitted ARQ blocks. However, the complexity of the RN increases.

In this paper, we classify the various ARQ mechanisms that have been proposed for multi-hop relay systems into end-to-end, hop-by-hop, and relay ARQ mechanisms and compare their performance. The modelling and analysis of these ARQ mechanisms are described in the following section.

III. MODELLING AND ANALYSIS OF ARQ MECHANISMS

A. System Model and Assumptions

To model the various ARQ mechanisms, we assume a two-hop chain topology as illustrated in Figs. 5. The \( c_1 \) and \( c_2 \) indicate the channel capacity between the source node and the RN and between the RN and the destination node, respectively, and have as a unit the number of packets which can be transmitted in one frame. The \( p_1 \) and \( p_2 \) indicate the packet error rate between the source node and the RN and between the RN and the destination node, respectively. We consider a frame structure that data is transmitted between the source node and the RN and between the RN and the destination node in one frame and the RN uses the next frame to relay the data.
received in the current frame. We assume that there are no feedback errors. The source node and the RN node have their own queue length, $q_{l1}$ and $q_{l2}$. We assume that packets arrive at the source node queue with a Poisson distribution at a rate $\lambda$. If the source node fails to transmit the packet to the RN, the packet is returned to the source node queue. If the RN fails to transmit the packet to the destination node, the packet is returned to the source node queue with a Poisson distribution at a rate $\lambda$. In the case of the hop-by-hop ARQ mechanism, the state transmission probability is the sum of one $\Lambda^+(\cdot)$ combination and many $\Lambda(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node. If $a_2$ is less than $q_{l1}$ and not less than $q_{l1} - \alpha$, this indicates that the queue of the source node can be full according to the returned packets, and hence, both $\Lambda$ and $\Lambda^+$ are used and the state transmission probability is the sum of many $\Lambda^+(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations.

In the case of the end-to-end ARQ mechanism, a state transition probability matrix, $P^e$, is calculated as follows. When $b_1 \leq \min[w, q_{l2} - c_1^2]$, $b_2 \leq \min[w_1, q_{l1} + b_1 - a_2]$, $c_1^2 = \min(c_1, w - b_1)$ and $0 \leq \alpha \leq c_1$ where $\alpha = b_1 - b_1 + \min(b_1, C_2)$, the state transmission probabilities are calculated as (4). $\alpha$ indicates the number of packets that the RN transmits successfully to the destination nodes. The state transmission probability is expressed in four different forms, according to the range of $a_2$. If $a_2$ is equal to $q_{l1}$, this indicates that the queue of the source node is full, and hence, only $\Lambda^+$ is used and the state transmission probability is the sum of many $\Lambda^+(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node.

In the case of the hop-by-hop ARQ mechanism, a state transition probability matrix, $P^h$, is calculated as (5), when $b_1 \leq w$, $b_2 \leq w$. The state transmission probability is expressed in three different forms, according to the range of $a_2$. If $a_2$ is not less than $q_{l1} - c_1$, this indicates that the queue of the source node can be full according to the returned packet, and hence, both $\Lambda$ and $\Lambda^+$ are used and the state transmission probability is the sum of one $\Lambda^+(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combination and many $\Lambda(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node.

In the case of the hop-by-hop ARQ mechanism, a state transition probability matrix, $P^h$, is calculated as (6), when $b_1 \leq w$, $b_2 \leq w$ and $c_1^2 = \min(c_1, w - b_1)$. For the relay ARQ mechanism, the state transmission probability has a form similar to that of the hop-by-hop ARQ mechanism, because neither mechanisms

![2-hop chain topology](image-url)

**B. Markov based Analytical Modelling**

To model the ARQ mechanisms, we use a Discrete Time Markov Chain (DTMC) technique. We use the following notations in the DTMC modelling:

- $a$: Number of queueing packets in the source node (integer in the range of $0-\alpha q_{l1}$)
- $b$: Number of queueing packets in the RS (integer in the range of $0-\alpha q_{l2}$)
- $(a, b)$: State of the DTMC model
- $P$: State transition probability matrix
- $P_{(a_1, b_1)-(a_2, b_2)}$: State transition probability from $(a_1, b_1)$ to $(a_2, b_2)$
- $w$: ARQ transmission window
- $\lambda$: Packet arrival rate in the queue of the source node
- $\Lambda(x)$: Probability that $x$ packets arrive in one frame

$$\Lambda(x) = \frac{e^{-\lambda} \lambda^x}{x!}. \quad (1)$$

- $\Lambda^+(x)$: Probability that more than $(x - 1)$ packets arrive in one frame

$$\Lambda^+(x) = 1 - \sum_{i=0}^{x-1} \frac{e^{-\lambda} \lambda^i}{i!}. \quad (2)$$

- $Q^{(1)}(y)$: Probability that $y$ packets are transmitted successfully in an $x$-packet transmission trial

$$Q^{(1)}(y) = \frac{x!}{(x-y)!} y^y \cdot (1-p)^{x-y} \cdot p^{x-y}. \quad (3)$$

where $p$ is the packet error rate.

The initial state of the DTMC model is $(0,0)$ and it transits to other states according to the arrival of new packets in the source node queue and the transmission of packets between the source node and the RN and between the RS and the destination node. So, the state transmission probability can be expressed by the combination of one $\Lambda(\cdot)$ (or $\Lambda^+(\cdot)$) function and two $Q^{(1)}(\cdot)$ functions. The $\Lambda(\cdot)$ (or $\Lambda^+(\cdot)$) function indicates the arrival of new packets in the source node and two $Q^{(1)}(\cdot)$ functions indicate the transmission of a packet by the source node and the RN.

In the case of the end-to-end ARQ mechanism, a state transition probability matrix, $P^e$, is calculated as follows. When $b_1 \leq \min[w, q_{l2} - c_1^2]$, $b_2 \leq \min[w_1, q_{l1} + b_1 - a_2]$, $c_1^2 = \min(c_1, w - b_1)$ and $0 \leq \alpha \leq c_1$ where $\alpha = b_1 - b_1 + \min(b_1, C_2)$, the state transmission probabilities are calculated as (4). $\alpha$ indicates the number of packets that the RN transmits successfully to the destination nodes. The state transmission probability is expressed in four different forms, according to the range of $a_2$. If $a_2$ is equal to $q_{l1}$, this indicates that the queue of the source node is full, and hence, only $\Lambda^+$ is used and the state transmission probability is the sum of many $\Lambda^+(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node. If $a_2$ is less than $q_{l1}$ and not less than $q_{l1} - \alpha$, this indicates that the queue of the source node can be full according to the returned packets, and hence, both $\Lambda$ and $\Lambda^+$ are used and the state transmission probability is the sum of many $\Lambda(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node. In the other cases, the state transmission probability is set to zero.

In the case of the hop-by-hop ARQ mechanism, a state transition probability matrix, $P^h$, is calculated as (5), when $b_1 \leq w$, $b_2 \leq w$. The state transmission probability is expressed in three different forms, according to the range of $a_2$. If $a_2$ is not less than $q_{l1} - c_1$, this indicates that the queue of the source node can be full according to the returned packet, and hence, both $\Lambda$ and $\Lambda^+$ are used and the state transmission probability is the sum of one $\Lambda^+(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combination and many $\Lambda(\cdot)Q^{(1)}(\cdot)Q^{(1)}(\cdot)$ combinations. Each combination varies according to the number of returned packets to the queue of the source node due to transmission failure between the RN and the destination node.
have any packets returned to the queue of source node from the
the queue of the RN. However, the relay ARQ mechanism uses
a single error recovery protocol that covers a complete multi-
hop route, unlike the hop-by-hop ARQ mechanism. Therefore,
the source node must transmit packets to the RN while taking
into consideration the packets that are queued in the RN.
As a result, \( c_1 \) is replaced by \( c_1' \), calculated as \( \min(c_1, w - b_1) \).

### C. Performance Analysis

We used delay and throughput as performance metrics for
the ARQ mechanisms. The delay is defined as the average
period from the arrival of a packet to its successful transmission
to the destination node. Throughput is defined as the average
number of packets that are transmitted to the destination node in
one frame.

Using the steady state probability matrix, \( \pi \), calculated
from \( \mathbf{P} \), the delay is calculated as follows. To use the
PASTA (Poisson Arrival See Time Average) theorem, we first
calculate an effective packet arrival rate in the source node
(\( \lambda_S \)) and the RN (\( \lambda_{RS} \)). The effective packet arrival
rate is defined as the packet arrival rate excluding any packets that
are dropped due to queue overflow. It is calculated as follows:

\[
\lambda_S = \lambda (1 - P_d^S), \quad \lambda_{RS} = \lambda_S (1 - P_{d_{RS}}).
\]

where \( P_d^S \) and \( P_{d_{RS}} \) indicate the packet drop probabilities in
the source node and the RN, respectively. \( P_d^S \) is calculated as follows:

\[
P_d^S = \sum_{i=0}^{q_1} QS(i) \sum_{j=0}^{\infty} q_1 i - j \lambda (j) \]

where \( QS(x) \) is the steady state probability that the number of
packets that are queued in the source node is \( x \). \( QS(x) \)
is calculated as \( \sum_{i=0}^{q_1} \pi_{(a,b)} \), where \( \pi_{(a,b)} \) is the steady state
probability of \( (a,b) \).

The packet drop probability in the RN, \( P_{d_{RS}} \), is calculated as follows:

\[
P_{d_{RS}} = \sum_{a=0}^{q_1} \sum_{b=0}^{q_2} \pi_{(a,b)} \times \sum_{t\in T_a} \max(0, t + b - q_2) \]

where \( T_a \) is defined as the set of the number of packets that
the source node can transmit to the RN when the number of
packets that are queued in the source node is \( a \).

Using (7)–(10), the delay (\( D \)) is calculated as follows:

\[
D = \frac{QLS}{\lambda_S} + \frac{QL_{RS}}{\lambda_{RS}}
\]

where \( QLS \) and \( QL_{RS} \) are the average queue lengths of
the source node and the RN, respectively. These can be calculated
easily, as follows:

\[
QLS = \sum_{a=0}^{q_1} \left[ a \times \sum_{b=0}^{q_2} \pi_{(a,b)} \right]
\]

\[
QL_{RS} = \sum_{b=0}^{q_2} \left[ b \times \sum_{a=0}^{q_1} \pi_{(a,b)} \right].
\]

Throughput is defined as the number of packets that the
RS transmits to the destination node successfully; hence, the
throughput (\( Thr \)) is determined by the state of the RN queue
and the packet error rate of the link between the RN and the
destination node. It is calculated as follows:

\[
Thr = \sum_{b=0}^{c_2} QR(b) \times (1 - p_2) \times b + \left[ 1 - \sum_{b=0}^{c_2} QR(b) \right] \times (1 - p_2) \times c_2
\]
where \( QR(x) \) is the steady state probability that the number of packets that are queued in the RN is \( x \). \( QR(x) \) is calculated as \( \sum_{a=0}^{\infty} \pi(a,x) \).

### IV. PROPOSED MODEL VALIDATION

In order to validate the proposed model, we simulated the ARQ mechanisms and compared simulation results with the numerical results. In all the simulations and numerical analysis, we assumed that \( q_1 \) was 40, \( q_2 \) was 20, \( w \) was 10, and \( c_1 \) and \( c_2 \) were 3. Figs. 6 shows a delay according to \( p_1 \) and \( p_2 \). In this simulation and numerical analysis, we assumed that \( \lambda \) was 0.5. From this figure, we can see that the end-to-end ARQ mechanism performs the worst and the hop-by-hop ARQ and relay ARQ mechanisms perform similarly. The delay increases exponentially as \( p_1 \) and \( p_2 \) increase and the simulation and numerical results match well. Figs. 7 shows throughput according to the packet generation rate. In this simulation and numerical analysis, we assumed that \( p_1 \) and \( p_2 \) were 0.512. From this figure, we can see that the end-to-end ARQ mechanism performs the worst and that the hop-by-hop ARQ performs the best and the simulation and numerical results match well. Figs. 8 shows the packet drop probability in the RN according to the packet generation rate and \( p_2 \). With the end-to-end ARQ and the relay ARQ mechanisms, the packet drop probability in the RN is zero, because the data transmission of the source is controlled by the ARQ transmission window, the size of which is less than the queue size of the RN. However, with the hop-by-hop ARQ mechanisms, the data is queued in the RN as the link condition between the RN and the destination node becomes worse than that between the source node and the RN and the traffic becomes heavier, because the data transmission of each link does not take into account the data transmission of the other link. We can check this tendency from this figure.

### V. CONCLUSIONS

We modelled various ARQ mechanisms for multi-hop relay systems considering continuous packet arrival, channel capacity, queue size, and ARQ window size. We analyzed the performance of the various mechanisms with respect to delay and throughput. We validated the proposed model by simulation.

### REFERENCES

2. IST-2003-507581 WINNER, “Description of identified new relay based radio network deployment concepts and first assessment by comparison against benchmarks of well known deployment concepts using enhanced radio interface technologies”, D3.2 version 1.0.