멀티미디어 Ad Hoc 무선망에서의 지연시간 보장 방안

양현호*

Providing Guaranteed Delay in Multimedia Ad Hoc Wireless Networks

Hyun-ho Yang*

요 약

멀티미디어 Ad Hoc 무선망은 이동하는 End User에게 멀티미디어 서비스를 제공할 수 있게 하는 유연한 해결방안을 제공하기 때문에 매력적인 주제이다. 그러나 Ad Hoc 무선망의 특별한 설계상의 문제점들로 인하여 자원을 공정하게 공유하면서 제한된 지연시간을 보장하는 것은 간단한 문제가 아니다. 이 논문에서는 지연시간보장 공정큐잉(DGFQ, Delay Guaranteed Fair Queueing) 방식을 분산적으로 구현하였다. 성능평가를 통하여 DGFQ 방식이 멀티미디어 Ad Hoc 무선망에서도 제한된 지연시간을 제어할 수 있음을 보였다.

ABSTRACT

The multimedia ad hoc wireless network is quite an attractive issue since it offers a flexible solution to enable delivery of multimedia services to mobile end users without fixed backbone networks. However, with the unique design challenges of ad hoc wireless networks, it is a non-trivial issue to provide bounded delay guarantee, with fair share of resources. In this paper, we implemented the delay guaranteed fair queueing (DGFQ) scheme distributively. Through the results of performance evaluation, we can conclude that DGFQ also performs well to control bounded delay in multimedia ad hoc wireless networks.

키워드

Fair queueing, Ad hoc network, Quality of Service (QoS), Multimedia network.

Ⅰ. Introduction

There are a series of wireless technologies newly emerging, e.g., Mobile Ad hoc Network (MANET), Bluetooth and sensor networks. These emerging wireless technologies are also required to provide a set of applications, e.g., both error-sensitive and delay-sensitive applications, over the bandwidth-constrained wireless medium. To fulfill this requirement, the issue of providing fair and delay bounded channel access among multiple contending hosts over a scarce and shared wireless channel is essential. Fair queueing has been a popular paradigm to achieve this goal in both wireline and packet cellular networking environments [1]-[5].

However, the problem of designing fully distributed, scalable, and efficient fair scheduling algorithms in the shared-channel ad hoc wireless network remains largely unaddressed. In essence,

---

* 순천청암대학 컴퓨터정보과학과

접수일자 : 2003. 6. 20

the unique characteristics of ad hoc wireless networks such as location-specific contention create spatial coupling effects among flows in the network graph, and the fundamental notion of fairness may require non-local computation among contending flows. Adding these features together, fair queueing in shared-channel multihop wireless environments is no longer a local property at each output link and has to exhibit global behaviors; this has to be achieved through distributed and localized decisions at each node.

In some related works the fair packet scheduling issues, on the aforementioned problems in ad hoc wireless networks, is addressed [6]-[8]. The focus of [6], [7] has been the problem formulation and an appropriate ideal centralized model for fair queueing in shared-channel multihop wireless networks. However the proposed distributed fair scheduling implementation can at best conceptually approximate the centralized model. In [8], they devised distributed and localized solutions such that local schedulers self-coordinate their local interactions to achieve the desired global behavior. They also propose a suite of fully distributed and localized fair scheduling models that use local flow information and perform local computations only. Though the contributions stated above, [8] mainly addressed on the fairness of the overall throughput performance for the various usage scenarios without consideration of the QoS factors such as delay performance especially for the multimedia wireless ad hoc wireless networks.

In our pervious work [9], we proposed a new fair queueing scheme i.e., delay guaranteed fair queueing (DGFQ), guaranteeing bounded delay of multimedia services. DGFQ scheme is basically a generalized processor sharing (GPS) based fair queueing scheme with some modifications to guarantee bounded delay. In detail, the *service differentiation coefficient* was introduced to apply additional weight factor for the delay guaranteed (DG) class over non-delay guaranteed (NG) class. With this policy, DGFQ provides better delay performance for DG class at the same fairness guarantee without serious increase of computational complexity. However, it was focused on the centralized network, rather than distributed one e.g., ad hoc wireless networks.

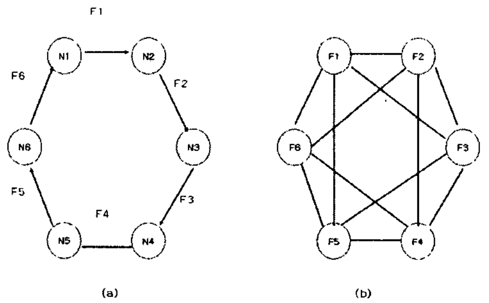
In this paper we implement the delay guaranteed fair queueing (DGFQ) [9] in the multimedia ad hoc wireless network using the distributed fair queueing protocol proposed in [8], to verify the controllability and adaptability of DFGQ on the bounded delay requirement in multimedia ad hoc wireless networks. From the results of performance evaluation, we can conclude that DGFQ also performs well to control bounded delay in multimedia ad hoc wireless networks.

The rest of the paper is organized as follows. Section 2 describes the network model for ad hoc fair scheduling. In section 3 we describe distributed implementation of delay guaranteed fair queueing (DGFQ) in the multimedia ad hoc wireless network. Section 4 presents a simulation-based performance evaluation of the implementation, and, finally in Section 5, we conclude our work.

II. Network Model

In this paper, we consider a packet-switched multihop wireless network in which the wireless medium is shared among multiple contending users, i.e., a single physical channel with capacity C is available for wireless transmissions. Transmissions are locally broadcast and only receivers within the transmission range of a sender can receive its packets. Each link layer packet flow is a stream of packets being transmitted from the source to the destination, where the source and destination are neighboring nodes that are within transmission range of each other. Two flows are contending with each other if either the sender or the receiver of one flow is within the transmission range of the sender or the receiver of the other flow [10]. We make the following assumptions [10]-[14]; (a) a collision occurs when a receiver is in the reception range of two simultaneously transmitting nodes, thus unable to cleanly receive signal from either of them; (b) we ignore capture effect in this work, (c) a node cannot transmit and receive packets simultaneously, and (d) neighborhood is a commutative property; hence, flow contention is also commutative. In addition, we do not consider non-collision-related channel errors. For simplicity of presentation, we only consider fixed packet size in this paper.

Flow Contention Graph The flow contention graph is introduced to describe the contending flows in the network [8]. The flow contention graph precisely characterizes the spatial domain, as well as the time domain, contention relationship among transmitting flows. In a flow graph, each vertex represents a backlogged flow, and an edge between two vertex denotes that these two flows are contending with each other. If two vertices are not connected, these two flows can transmit simultaneously, thus spatial reuse is possible. Therefore, the flow graph explicitly describes which flows are contending and which flows can be concurrently transmitting.



(a) (b)

Fig. 1. Node Graph and Flow Graph in Location Dependent Contention; (a) Original Node Topology Graph (b) Flow Graph.

As an example, Fig. 1(a) shows the node topology graph consisted of 6 nodes. In the figure, any two nodes connected by a vertex are in the communication range. Thus, for example,

when node N1 transmits a packet of flow F1, node N2 should be remained receiving mode, moreover, if node N3 transmits a packet of flow F3, collision may occur in the node N2. Therefore F2, F3 are contending, further, neighbouring flows of F1 and in the similar reason, so do F5 and F6. Fig. 1(b) shows the flow contention graph for the six flows in the node graph. Each node in an ad hoc wireless network maintains information for flows within one-hop neighborhood in the flow contention graph. However, one hop neighborhood in a flow graph will translate to the two-hop neighborhood in the real node graph in practice. In Fig. 1(b), one-hop neighborhood of flow F1 includes F2, F3, F5, F6. Therefore, for a given flow *f*, it is required to maintain flow information for flows that are within the transmission range of either f's sender or its receiver. However, for any given node, our goal is to maintain flow information (i.e., service tags) for flows only within its one-hop neighborhood in the node graph. That is, no node needs to be aware of flow information at nodes that are more than one hop away in the node graph.

III. Distributed Implementation of DGFQ in Ad Hoc Wireless Networks

In this section, we describe the implementation of DGFQ in a distributed ad hoc wireless network. We first explain again the DGFQ scheme briefly, then explain basic scheduling operations and the protocol for a distributed implementation.

3.1. Delay Guaranteed Fair Queueing(DGFQ)

DGFQ basically adopts start-time fair queueing (SFQ) algorithm proposed in [4]. In DGFQ, as is in the SFQ, twotags i.e., a start tag and a finish tag, are associated with each packet. However, unlike weighted fair queueing (WFQ) and start-time fair queueing (SCFQ), packets are scheduled in the increasing order of the start tags of the packets. Furthermore, v(t) is defined as the start tag of the packet in service at time t. Finally, we assume that, in SFQ, WFQ or DGFQ scheme, there is a certain interval of time in which all flows are scheduled at least once, we call it scheduling interval.

All flows are classified into a number of classes according to their delay bound requirements. The simplest and basic classification is to make two classes, one for delay guaranteed (DG) flows and the rest for non delay guaranteed (NG) flows. In our scheme, we introduce the service differentiation coefficient, α ($0 < \alpha \leq 1$), to handle each flow classes differently. When $\alpha = 1$, which is the case for NG class, our proposed scheme is identical to SFQ. By varying α, we can customize delay bound for individual flows i.e., adjust the relative service order of each flows in a scheduling interval.

The complete algorithm is defined as follows.

i) On arrival, k^{th} packet of flow f, p_f^k, is stamped with start tag $S(p_f^k)$, computed as

$$S(p_f^k) = \max[A(p_f^k)], F(p_f^{k-1}), \quad k \geq 1 \tag{1}$$

where $A(p_f^k)$ is the arrival time of packet p_f^k and $F(p_f^k)$ is the finish tag of packet, p_f^k defined as

$$F(p_f^k) = S(p_f^k) + \alpha_f \frac{l_f^k}{\phi_f} \tag{2}$$

where $F(p_f^0) = 0$ and ϕ_f is the weight of flow f, l_f^k is the length of packet p_f^k, and α_f ($0 < \alpha_f \leq 1$) is the service differentiation coefficient for flow f. The value of α_f is 1 for NG class, or appropriate value for DG class.

ii) Initially the system virtual time is 0. During a busy period, the system virtual time at time t, $v(t)$, is defined to be equal to the start tag of the packet in service at time t. At the end of a busy period, $v(t)$ is set to the maximum of finish tag assigned to any packets that have been serviced by then.

iii) Packets are serviced in the increasing order of the start tags; ties are broken arbitrarily.

3.2 Basic Scheduling Operations

The detailed operations for distributed implementation of delay guaranteed fair queueing (DGFQ) in multimedia ad hoc wireless network consist of the following four parts:

- Local state maintenance: Each node n maintains a local table E_n, which records each flow's current service tag for all flows in its one-hop neighborhood of the flow graph. Each table entry has the form of $[f, T_f]$, where T_f is the current service tag of flow f, e.g., the most recent start tag of flow *f*.

- Tagging operations: Two tags, i.e., a start tag and a finish tag, are assigned for each arriving packet, using DGFQ algorithm described in the previous Section 3.1, for each flow f in the local table.
- Scheduling loop: After the tagging operation, at the sender node n of a flow f, the following procedure is performed, whenever the node n hears that the channel is clear,

(a) if the flow f has the smallest service tag in the table E_n, of node n, transmit the head-of-line packet of flow f immediately;

(b) otherwise, set the backoff timer B_f of flow f as

$$B_f = \sum_{g \in S} I(T_{g(t)} < T_f(t)) \quad slots$$

where S is the set of backlogged flows in the system and I(x) denotes the indicator function, i.e., I(x) = 1, if x>0; I(x) = 0, otherwise.

(c) if flow f 's backoff timer expires and the channel is idle, transmit the head-of-line packet of flow f.

- Table updates: whenever node n hears a new service tag $T_{g'}$ for any flow g on its table E_n, it updates the table entry for flow g to $[g, T_{g'}]$. Whenever node n transmits a head-of-line packet for flow f, it updates flow f 's service tag in the table entry.

We provide an illustrative example to show how the algorithm works. In the example, as shown in Fig. 2, four flows are scheduled from the sender node to its respective receiver node and the dotted line denotes the two nodes are within the communication range. It is assumed that the initial virtual time V = 0, and the initial service tags for the four flows are T1 = 1, T2 = 2, T3 = 3, T4 = 4. The table maintained at each sender of the four flows and the backoff calculation and table updates before and after transmission of flows 1 and 4 are also shown in Fig. 2.

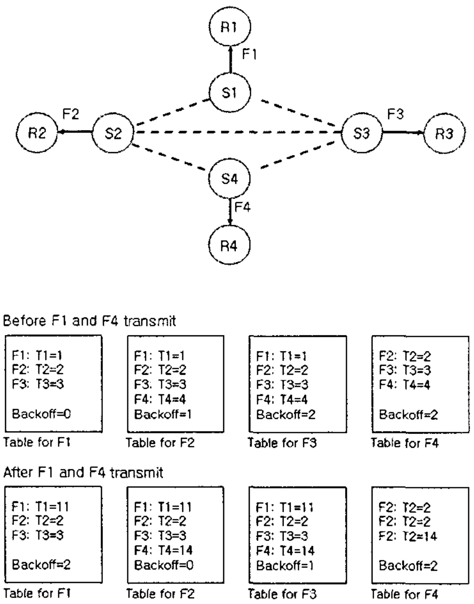


Fig. 2. An Illustrative Example of Node Graph and Table Updates

3.3. Message Sequence

In the distributed implementation protocol, each data transmission follows, either controlled by DGFQ or not, the basic sequence of RTS-CTS-DS-DATA-ACK handshake is applied after backoff of certain number of time slots.

When a node has a packet to transmit, it waits for an appropriate number of slots before it initiates the RTS-CTS handshake. In particular, the node checks its local table and sets a backoff timer for flow *f* to be the number of flows with tags smaller than the tag of flow *f*. This way, the local minimum-tag flow backs off for zero slot and contends for the channel immediately.

If the backoff timer of f expires without overhearing any ongoing transmission, it starts RTS carrying the backoff according to the table at the receiver's side, denoted as B_f^R, to initiate the handshake. If the node overhears some ongoing transmission, it cancels its backoff timer and defers until the ongoing transmission completes. In the meantime, it updates its local table for the tag of the on-going neighboring transmitting flow. When other nodes hear a RTS, they defer for one CTS transmission time to permit the sender to receive a CTS reply.

When a receiver receives a RTS, it checks its local table. If B_f^R is greater than or equal to the backoff value for flow f in the receiver's local table, it responds with CTS. Otherwise, the receiver simply drops RTS. This procedure is required for maintaining the table information at both sender and receiver nodes. Detailed mechanism descriptions are given in [8].

Once a sender receives the CTS, it cancels all remaining backoff timers for other flows and transmits DS. When hosts hear either a CTS or a DS message, they will defer until the DATA-ACK transmission completes.

Propagating Service Tag Update In order to propagate a flow's service tag to all its one-hop neighbors in the node graph and reduce the chance of information loss due to collisions during this service tag information propagation, the tag T_f for flow f is attached in all four packets RTS, CTS, DS and ACK, i.e., the old tag in RTS and CTS packets, and updated tag in DS and ACK packet.

Ⅳ. Performance Evaluation

In this section, we evaluate the performance of the distributed implementation of DGFQ in multimedia ad hoc wireless network. In the following we describe the simulation environment and discussions on the results for the performance measures such as throughput, average delay, and maximum delay.

4.1 Simulation Environment

We use simulations to evaluate the performance of our distributed implementation of DGFQ in multimedia ad hoc wireless networks. The following is the simulation environment used in this simulation.

The radio model is based on existing commercial wireless networks with radio transmission range of 250 meters and channel capacity of 2Mbit/sec which is typical capacity of current wireless mobile networks. Moreover, for the distributed implementation of DGFQ scheme, error free channel model is assumed to concentrate our evaluation work on the key features of proposed scheme, i.e., the controllability and adaptability of DGFQ scheme in distributed network environment such as multimedia ad hoc wireless networks to provide delay guaranteed service.

As the traffic source model, we choose the modified MPEG source, described in [15]. Moreover, we assumed that all the sources have identical characteristics. In this video sourec model, there are three types of frame, i.e., I, B and P frames. Each frame size is determined by a Lognormal distribution with a specified mean and standard deviation. A video source generates 24 frames per second.

Further, we adopt the simulation scenario 3 used in [8], which includes 14 nodes transmitting 10 flows, because it is a reasonable scale considering our target environment of multimedia ad hoc wireless networks. Fig. 3.

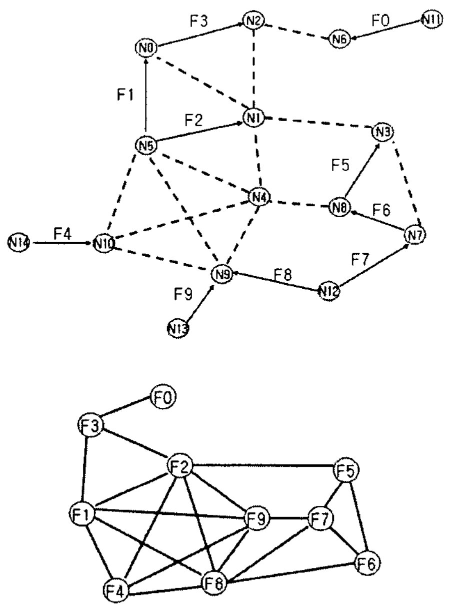


Fig. 6. Node Graph and Flow Graph of Simulated Multimedia Ad Hoc Wireless Network.

shows the node graph and flow graph of simulated network respectively.

More specifically, flow $F4$ is controlled with the aforementioned service differentiation coefficient, α to testify the controllability of DGFQ for guaranteed delay provision in distributed network environment. For all other flows in the simulation, α is fixede to the value of 1. In addition, the simulation results for flow F4 are compared with that of other contending flows and overall average.

Finally, each simulation is run for 1000 seconds, and we selected average delay, maximum delay and throughput as the performance measures as in [9]. Detailed definitions and discussions for these measures are described in the following Section 4.2.

4.2. Results and Discussions

The discussions on the simulated results for the distributed implementation of DGFQ is given below, specifically for the performance measures such as throughput, average delay, and maximum delay.

4.2.1. Throughput

We used throughput as a fairness measure, which is total transmitted packets during the whole simulation duration, say, 1000 seconds. Fig. 4 shows the throughput of flows with scattered points and their regression. As reported in [9], basically there is only a minor differences in throughput between flows either controlled (F4) or not (all other flows). In the figure, the thick solid line represents the average throughput of all flows, and the thick dashed line shows the throughput of the controlled flow (F4).

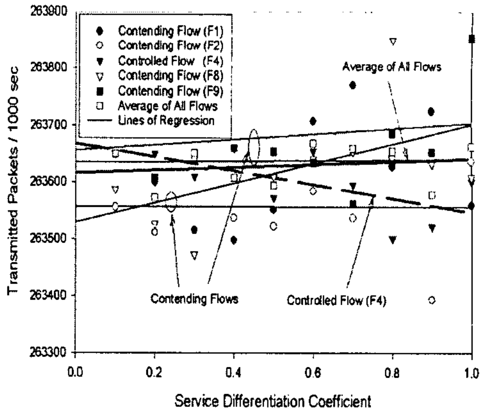


Fig. 7. Throughput

In particular, the number of transmitted packets of F4 is increasing with decreasing α value, It is because α controls F4 with the share of channel in some extent. In detail, α value of all contending flows are 1, as mentioned in the simulation environment description, when $\alpha=1$, flow F4 shares the channel resources with all other flows by same policy. On the contrary, with varying the α value, F4 get more weight for the share of resources to transmit packets, even though the difference is minor. It should also be noticed that it is possible to control individual flow with varying α.

4.2.3. Average delay

In our work average delay is defined as the average time interval between the arrival and departure of a packet for a certain time duration. As shown in the Fig. 5, the service differentiation coefficient α is the key parameter to manage delay performance. Again, in the figure, thick solid line represents the overall average delay, averaged for all flows, and thick dashed line shows the average delay of the controlled flow (F4). With varying α we can control the average delay of flow F4. As shown in the figure, we can notice that the average delay of the controlled flow (F4) is proportional to α while all other flows are not.

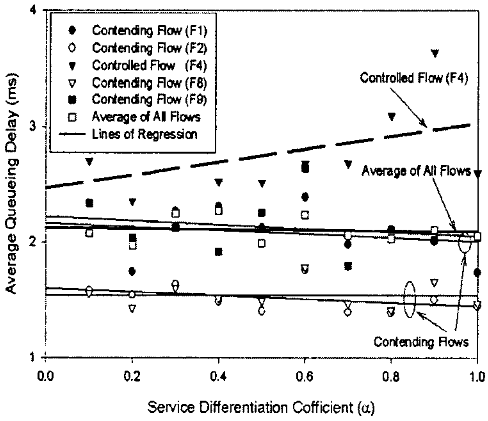


Fig. 5. Average Delay

4.2.3. Maximum delay

The maximum delay is another critical performance measure for real time multimedia flows. We define maximum delay as the maximum interval between the arrival and departure of a packet in the system in a certain duration of time, say, simulation duration. We can get the results simultaneously with average delay from the same simulation. As in the previous figures, in Fig. 5, thick solid line represents the overall maximum delay, averaged for all flows, and thick dashed line shows the maximum delay of the controlled flow (F4). From the Fig. 5, we can conclude that maximum delay could also be controllable with α, which means DGFQ controls the maximum delay also in distributed networks. Maximum delay is more tightly controlled than average delay, though both of them are fall in to the range of required bound for typical realtime video applications.

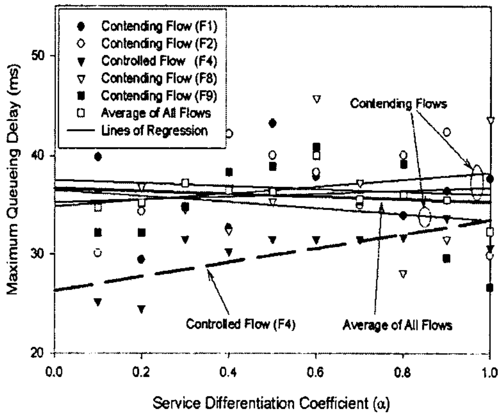


Fig. 6. Maximun Delay

V. Conclusion

We implemented a delay guaranteed fair queueing scheme, DGFQ [9], distributively in the multimedia ad hoc wireless network environment. As far as throughput is concerned, there is only a minor difference in throughput between flows either controlled by service differentiation coefficient (α) or not. Through the simulation results, average delay and maximum delay are controlable in our implementation. In summary, the controllability and adaptability of DGFQ on the multimedia traffic in the distributed network environment was verified.

We just considered about a limited network environment, i.e., stationary nodes with error-free wireless channel, which is too idealistic to apply our work in the practical systems. So, much more work is needed to be done for the dynamic topology variation by mobile nodes in error-prone wireless channel case as a future work.

References

[1] A. Demers, S. Keshav, and S. Shenker, "Analysis and simulation of a fair queueing algorithm," Proc. ACM SIGCOMM, pp. 1-12, Sep. 1989.

[2] A. K. Parekh and R. G. Gallager, "A generalized processor sharing approach to flow control in integrated services networks: the single-node case," IEEE/ACM Transactions on Networking, Vol.1, No.3, pp. 344-357, Jun. 1993.

[3] J. C. R. Bennett and H. Zhang, "WF2Q: Worst-case fair weighted fair queueing," IEEE INFOCOM' 96, 1996.

[4] P. Goyal, H.M. Vin and H. Chen, "Start-time fair queueing: A scheduling algorithm for integrated service access," ACM SIGCOMM' 96, Aug. 1996.

[5] S. Lu, V. Bharghavan and R. Srikant, "Fair scheduling in wireless packet networks," IEEE/ACM Trans. Networking, August 1999.

[6] N. H. Vaidya, P. Bahl ann S. Gupta, "Distributed fair scheduling in a wireless LAN," ACM MOBICOM'00, August 2000.

[7] H. Luo and S. Lu, "A topology-independent fair queueing model in ad hoc wireless networks," IEEE ICNP'00, Nov. 2000.

[8] H. Luo, P. Medvedev, J. Cheng and S. Lu, "A Self-Coordinating Approach to Distributed Fair Queueung in Ad Hoc Wireless Networks," IEEE INFOCOM'91, August 1991.

[9] H. Yang and K. Kim, "Delay Guaranteed Fair Queueing (DGFQ) in Multimedia Wireless Packet Networks," Lecture Notes in Computer Science (LNCS2839), Springer-Verlag, pp. 170-182, Sep., 2003.

[10] V. Bharghavan, A. Demers, S. Shenker, and L. Zhang, "MACAW: A medium access protocol for wireless LANs," ACM SIGCOMM'94, 1999.

[11] I. Chlamtac and A. Lerner, "Fair algorithms for maximal link activation in multihop radio networks," IEEE Trans. Communications, 35(7), July 1987.

[12] J. Ju and V. O. K. Li, "An optimal topology- transparent scheduling method in multihop packet radio networks," IEEE /ACM Trans. Networking}, 6(3), June 1998.

[13] I. Chlamtac and A. Farago, "Making transmission schedules immune to topology changes in multi-hop packet radio networks," IEEE/ACM Trans. Networking}, 2(1), February 1994.

[14] Z. Tang and J.J. Garacia-Luna-Aceves, "A protocol for topology-dependent transmission scheduling in wireless networks," WCNC'99, September 1999.

[15] S. Lee, K. Kim, A. Ahmad, "'Delay and data rate decoupled fair queueing for wireless multimedia networks," Proc. GLOBECOM '02, vol. 1, pp.946-950, 2002.

저자 소개

양현호(Hyun-ho Yang)

1986년 광운대학교 전자공학과 (공학사)
1990년 광운대학교 대학원 전자공학과 (공학석사)
2003년 광주과학기술원 정보통신공학과 (공학박사)
1989년 ~ 1990년 삼성 SDS 주식회사 Network Engineer
1991년 ~ 1997년 포스데이타 주식회사 POSCO 광양제철소 Network Operation Manager
1997년 ~ 현재 순천청암대학 컴퓨터정보과학과 조교수
※ 관심분야 : 무선멀티미디어 통신, Ad Hoc 네트워크