行政院國家科學委員會專題研究計畫　期中進度報告

無線隨意網路整合頻道補償暨電源控制的排程機制之研究

(1/2)

計畫類別： 個別型計畫
計畫編號： NSC94-2213-E-009-115-
執行期間： 94 年 08 月 01 日至 95 年 07 月 31 日
執行單位： 國立交通大學資訊科學學系(所)

計畫主持人： 趙禧綠

報告類型： 精簡報告
報告附件： 出席國際會議研究心得報告及發表論文
處理方式： 本計畫可公開查詢

中　華　民　國 95 年 5 月 29 日

摘要

網際網路已成為目前最常被使用的通訊架構，而為了能在網際網路上提供具不同服務品質的服務，方法之一是藉助排程機制的支援。本計劃的第一年之研究主題是針對無線隨意網路，設計考量頻道狀況的排程機制。我們提出兩套機制，一是以累積頻寬為主(稱為CBCP)，一是以服務時間戳記為主的協定(稱為 TBCP)。經由實驗證明，我們所提出的兩個機制可以保證對使用頻寬有最小要求的資料流之服務品質、提高整體公平性、同時增加網路整體輸出總量。

關鍵詞：無線隨意網路，頻道補償，公平排程機制

Abstract

As the Internet becomes a global communication infrastructure and users' desires to run real-time multimedia applications, scheduling mechanism is one mechanism used to provide Quality of Service (QoS) to different flows. The first year research topic of this project is to propose a fair scheduling mechanism with channel compensation for mobile ad hoc networks. Two scheduling algorithms are proposed: one is CBCP, which is credit-based, and the other is TBCP, which is timestamp-based. From the simulation results, both CBCP and TBCP can guarantee QoS flows' minimum bandwidth requirements, improve the fairness of residual bandwidth allocation, and increase the overall network throughput.

Keywords- mobile ad hoc networks, channel compensation, fair scheduling,

I. Introduction

An ad hoc network is a self-organizing wireless network comprised only of mobile nodes without the support of any pre-existing wired infrastructure. According to different decision metrics, existing work for fair scheduling in ad hoc networks can be classified into two categories: *timestamp based* [1,2] and *credit based* [3]. The timestamp mechanisms in [1] and [2] work similarly. Using [2] as an example, each arriving packet is locally assigned two timestamps: a start tag and a finish tag. Either timestamp can be chosen as the service tag. The packet with the smallest service tag will be sent first. In [3], scheduling is based on the credit value. The unused credit can be accumulated for future use. Each flow is associated with three parameters: a credit, a usage, and an excess, where excess = usage – credit. The one with the smallest excess value has the priority to transmit.

All existing work of fair scheduling in ad hoc networks [1,2,3] simply assumes channels are error-free. This assumption, however, is unrealistic for wireless networks, which usually suffer from high error rates, such as location-dependent errors or bursty errors. In this paper, we will investigate the impact of channel errors on fair scheduling and discuss the channel compensation issue of fair scheduling in ad hoc networks.

A. Related Work for Fair Scheduling with Error Compensation in Wireless Networks

Existing work proposed to achieve fairness with error compensation for wireless networks [4] are all based on the support of base stations and work for one-hop wireless channels.

In [5], the Channel-condition Independent packet Fair Queueing (CIF-Q) is proposed. Each session maintains a parameter called *lag* to indicate if it should be compensated. If a session is not leading and the channel is error-free at its scheduled time, its head-of-line packet is transmitted; otherwise, the time slot is released to other sessions. The problem with CIF-Q is that leading sessions are not allowed to terminate unless all their leads have been paid back, regardless of whether such terminations are caused by broken routes. This property makes CIF-Q an infeasible solution for ad hoc networks, because a connection may be broken due to node movements.

In [6], the Idealized Wireless Fair-Queueing (IWFQ), and the Wireless Packet Scheduling protocol (WPS) are proposed. According to the weight of each session, the base station calculates the number of slots per frame each session can use. If a session experiences channel errors at its scheduled time, the base station first determines if any other sessions can exchange their slots with this session within the same frame; otherwise, the base station will compensate the error slots of this session in a later frame. Again, this mechanism is not suitable for ad hoc networks, due to the need of support from base stations.

In [7], a virtual compensation session is introduced into the scheduling process. The virtual session is a session, which always experiences an error-free channel at its scheduled time, but it does not really generate packets. The slots received by this virtual session are used for compensation, i.e., they will be reassigned to lagging sessions. Since this mechanism only works for single-hop wireless networks, again, it is not suitable for multihop wireless ad hoc networks.

In [8], the Bandwidth-guaranteed Fair Scheduling with effective excess bandwidth allocation (BGFS-EBA) is proposed. In BGFS-EBA, a flow can transmit packets while its channel condition being in bad state by means of splitting a packet into several low-rate packets with enhanced error-correction mechanism. Besides, the excess bandwidth that some flows do not use up will be shared by all lagging flows. The scheduler maintains a dummy flow, which does not have actual packets to be sent, to fill up the bandwidth. Besides, each flow maintains a parameter, called deadline, to be the scheduling metric. BGFS-EBA maintains an idealistic full-load error-free system. The scheduling process has two phases: the first phase is for the scheduler to assign a slot to the flow with earliest deadline; the second phase is for the scheduler to further decide which real flow is qualified to use the slot. However, the authors do not describe detailed flow weight assignment. Furthermore, BGFS-EBA is designed for base station support environment.

B. Problem Statement

In this paper, we will study fair scheduling in ad hoc networks in which channel errors are considered. In particular, we will focus on the timestamp-based fair scheduling mechanism.

There are two timestamp-based error-free mechanisms in the literature proposed for ad hoc networks (i.e., [1,2]). However, these two protocols only work for single-hop flows, a special case in ad hoc networks. In the next section, we will show how to extend these single-hop timestamp mechanisms to serve multihop wireless networks.

The rest of this report is organized as follows. Sec. II describes the proposed timestamp-based protocol. Sec. III shows the simulation results. Finally, the report is concluded in Sec. IV.

II. Timestamp-Based Compensation Protocol (TBCP)

In this section, a timestamp-based fair scheduling mechanism, called Timestamp-Based Compensation Protocol (TBCP), is proposed for multihop wireless networks, under which channel errors are considered. TBCP adopts Start-time Fair Queueing (SFQ) [9] as its scheduling discipline and selects the start tag as its service tag.

A. System Model

We assume a TDMA-based system operates over a single channel shared by all hosts, and the bandwidth is represented as frames of time slots. Each frame has several slots: some are control slots and some are data slots. A valid route should be constructed, using whatever ad hoc routing mechanisms, before data packets are transmitted. Note that the spatial channel reuse mechanisms in [1,2] are still applicable to our mechanism.

Each node periodically measures the Signal-to-Noise Ratio (*SNR*) on the channel to determine the channel state. When the *SNR* value falls below a predefined threshold, the channel status is deemed error-prone and the node is prevented from transmitting packets temporarily. If the *SNR* value exceeds the predefined threshold again, the node is allowed to transmit again.

B. Normalized Flow Weight

Assume that there are *n* flows passing through node *N*. Let x_i^N denote the bandwidth share of flow *i* at node *N* (called the flow weight of flow *i* at node *N* in this paper), and is expressed as

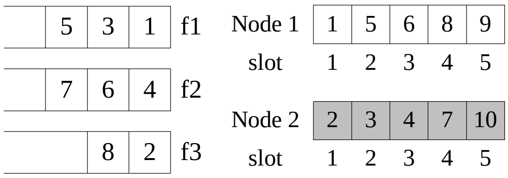
$$x_i^N = \frac{1 - \sum_{j=1}^{n} Resv_j}{n} + Resv_i \, , \tag{1}$$

where $Resv_i$ is the minimum bandwidth requirement of flow *i*, represented as a fraction of the channel bandwidth. Flow weight x_i^N indicates the fraction of channel bandwidth used by flow *i* at node *N*, so as to fairly share the residual bandwidth of node *N* and meet the minimum bandwidth requirement of flow *i*. For example, assume that there are three flows, say, f_1, f_2, and f_3 passing through node *N*. Flows f_1 and f_2 are guaranteed flows, each with a *Resv* value of 0.2; flow f_3 is a best effort flow. Based on (1), the flow weights of the three flows at node *N* are (x_1^N, x_2^N, x_3^N)=(0.4, 0.4, 0.2).

In TBCP, the actual bandwidth share for each flow at node *N* is determined by the flows at all nodes in the interference range (i.e., nodes located within two hops), not just solely depending on those flow passing through node *N*. Due to lack of coordination for transmissions, nodes located in the interfering range contend the channel, and should be considered in the calculation of bandwidth share for each flow. To reflect this fact, the normalized flow weight of each flow is defined as follows.

Let F_N and $F_{N'}$ be the set of flow segments passing through node *N* and *N'*, respectively, where *N'* is within 2-hop range of node *N*. Besides, let S_N be the set of node including node *N* and all other nodes in its interfering range (i.e., two hops). For each flow segment *f* at node *N*, its normalized flow weight is expressed as.

$$\omega_f^N = \frac{x_f^N}{\sum_{i \in (F_N \cup F_{N'}), j \in S_N} x_i^j} \tag{2}$$



(a) (b)

Figure 1. An example to explain TBCP.

This normalized flow weight ω_f^N is then used for the calculation of service tag of flow f at node N. The number of slots per frame that flow f can obtain at node N is the multiplication of ω_f^N and the number of slots per frame. Note that since the calculations of (1) and (2) are performed independently at each node, different single-hop flow segments of a multihop flow may have different bandwidth shares and normalized flow weights.

For example, considering two nodes, say N_1 and N_2, are in an ad hoc network. These two nodes are within the transmission range of each other. Let $Resv_i^j$ indicate the *Resv* value of the i^{th} flow at node j. The flow information of N_1 is $\left(Resv_1^{N_1}, Resv_2^{N_1}, Resv_3^{N_1}\right)$= (0.2,0.2,0), and that of N_2 is $\left(Resv_1^{N_2}, Resv_2^{N_2}, Resv_3^{N_2}\right)$= (0.4,0,0). Thus the flow weight calculated by N_1 and N_2 are (0.4,0.4,0.2) and (0.6,0.2,0.2), respectively, and the normalized flow weights of all flows are $\left(\omega_1^{N_1}, \omega_2^{N_1}, \omega_3^{N_1}, \omega_1^{N_2}, \omega_2^{N_2}, \omega_3^{N_2}\right)$= (0.2, 0.2, 0.1, 0.3, 0.1, 0.1).

C. TBCP Operations

C.1 Transmission Order Determination at a Node

The transmission order of a packet is determined with three factors: the service tag of the packet, the number of slots per frame a flow can use, and flow's *Q-size*. For example, the numbers in Fig. 1 show the packets' service tags of three flows f_1, f_2, and f_3 at node N. Assume that each frame is comprised of five slots, and f_i^j indicates the j^{th} packet of flow i. Besides, the number of slots per frame each flow can use is (2,2,1). The transmission order of packets at node N in Fig. 1 is then $< f_1^1, f_3^1, f_1^2, f_2^1, f_2^2 >$. Note that since f_1 has already used two slots, the last slot of this frame is assigned to f_2 even though the service tag of f_1^3 is smaller than f_2^2.

C.2 Message Exchange

Each node exchanges the information about the transmission order of packets determined at step 1) with its neighbors. Thus each node knows the service tags of other nodes, and also learns when it will transmit packets. For example, assume that there are two nodes in an ad hoc network, and the corresponding transmission order with the service tag is shown in Fig. 1. Let N_i^j indicates the j^{th} slot of node i. The transmission sequence is $< N_1^1, N_2^1, N_2^2, N_2^3, N_1^2 >$, $< N_1^3, N_2^4, N_1^4, N_1^5, N_2^5 >$. Note that if multiple nodes have packets with the same service tag, the tie will be broken based on their node IDs.

C.3 Channel Error Handling

Each node keeps monitoring its channel state. When the channel is error-prone, the node stops exchanging transmission messages with its neighbors. Once the channel recovers, the error-prone node resumes the exchanges. Consequently, if this node has packets with service tags smaller than its neighbors after recovery, these packets still have higher priority to be transmitted. In Fig. 1(b), if node 2 experiences an error-prone state at slot 4, the modified transmission sequence is $< N_1^1, N_2^1, N_2^2, X, N_1^2 >, < N_2^3, N_1^3, N_2^4, N_1^4, N_1^5 >$.

D. Multihop Flows

In ad hoc networks, each mobile node acts as both a router and a host. Thus, a multihop flow can be modeled as multiple single-hop flows, and each node schedules the single-hop flows passing through it independently. It is insufficient for a node to schedule for multihop flows, relying only on the selected scheduling parameter. To solve this problem, a new parameter called *Q-size* is defined. The *Q-size* of a flow is used to indicate the number of packets received from its previous hop and waiting to be sent to the next hop. In other words, those single-hop flows belonging to the same multi-hop flow are correlated with the *Q-size* parameter. Therefore, among all nodes with nonzero *Q-size* values, the node with the least scheduling parameter value can use the next time slot to transmit.

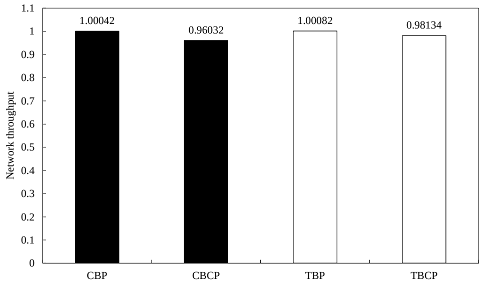
III. Simulation

In this section, we will provide simulation results to evaluate the performance of the fair scheduling mechanisms with and without considering channel errors. The simulation environment is described as follows. 20 mobile nodes are randomly distributed in a 1000-meter by 1000-meter area. The transmission range of each node is set to 250 meters. We randomly select nodes, some as flow sources and some as flow destinations. Each flow may either be a best effort or guaranteed flow and is continuously with backlogged packets. Each packet is assumed to occupy one time slot, and has fixed packet length. The mobility pattern of each node follows the modified random waypoint model. The simulation time lasts 50,000 slots.

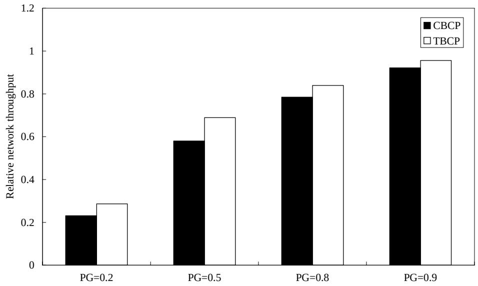
The wireless channel is modeled as a two-state discrete Markov chain in our simulation, as in [10]. Let p_g be the probability that the next time slot is in the good state given that the current time slot is error-prone, and let p_e be the probability that the next time slot is error-prone given that the current slot is in the good state. State G means the channel is good, and state E means the channel is error-prone. Then the steady state probabilities P_G and P_E in the good and error-prone states, respectively, are given by

$$P_G = p_g / p_g + p_e \quad \text{and} \quad P_E = p_e / p_g + p_e \,.$$

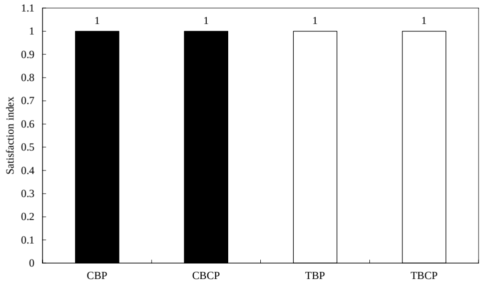
In this simulation, we compare TBCP with TBP (the error-free model of TBCP, i.e., assuming that channels are error-free in the scheduling). Besides, we implement the spatial channel reuse scheme as in [2]. For comparison purpose, we also show the performances of credit-based protocols—CBCP and CBP [3] (the error-prone and error-free model, respectively). CBCP differs from CBP in that when a node detects an error-prone state at its scheduled time, the node will inform its scheduler to stop assigning further slots to it. The Credit value of this node at the scheduler continues to be accumulated. Thus, once the channel has recovered, the node has a small Excess value as compared to other error-free nodes, which allows this node to have priority



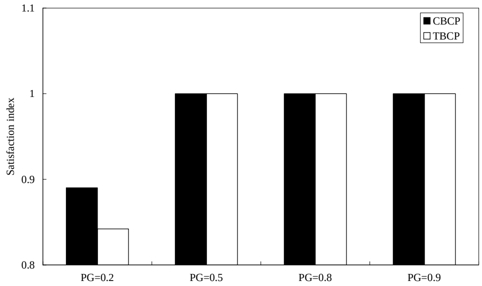
(a) Network throughput



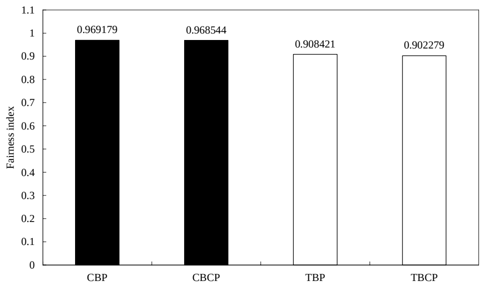
(d) Relative network throughput



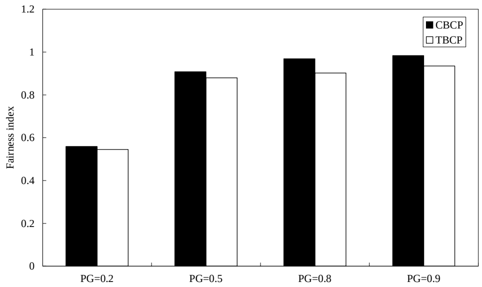
(b) Satisfaction index



(e) Satisfaction index for different P_G values



(c) Fairness index



(f) Fairness index for different P_G values

Figure 2. Two-state Markov-chain error model

to obtain slots. This node then in turn assigns this slot to the flow with the least Excess value among all nonzero *Q-size* flows.

The performance metrics measured in the simulation include the network throughput (ρ), satisfaction index, and fairness index, as defined in [11]. Besides, we define a new parameter, called relative network throughput (ψ). This parameter is to evaluate the performance of each approach considering channel errors and is compared with error-free system, and is defined as ψ_{CBCP}/ψ_{CBP} and ψ_{TBCP}/ψ_{TBP} for credit-based and timestamp-based mechanism, respectively.

We generate five multihop flows: three guaranteed flows and two best-effort flows. The steady state probabilities P_G and P_E are 0.8 and 0.2, respectively. The Min and Max speed are set to be 10 meters per second and 30 meters per second, respectively. The network throughput of each approach is shown in Fig. 2(a). The satisfaction and fairness indices of each approach are shown in Figs. 2(b) and (c), respectively. We find that mobility causes broken routes more frequently, thus degrades network throughput. However, node mobility may increase the probability for a flow to change the area in which it is located. This helps improve global fairness. Finally, we study the impact of channel errors on the proposed approaches. We vary the value of P_G but fix all other parameters. The relative network throughput of each approach is shown in Fig. 2(d). As the value of P_G increases, the relative network throughput increases. The reason is that a larger P_G value means more good slots, leading to more successful transmissions and higher network throughput. The satisfaction and fairness indices of each setting are shown in Figs. 2(e) and 2(f), respectively. Since a large P_G value makes more good slots, the possibility to satisfy QoS flows is higher. Thus, it gives a higher satisfaction index. A large P_G value also increases the opportunity for a node to be compensated after its channel recovers, and thus has a higher fairness index.

IV. Conclusion

In this report, we discuss the channel compensation issue of fair scheduling and propose one such mechanism for mobile multihop networks, which is timestamp-based, called TBCP, and is based on a virtual clock mechanism. The proposed mechanism supports multihop flows, and performs well when node mobility is supported. We describe the detailed operations of TBCP, and conduct simulations to evaluate the performance. From the simulation results, we demonstrate that the proposed mechanism satisfies QoS flow demands and provides global fairness for best effort flows. Finally, we also analyze the flow throughputs of TBCP, and verify the analytical result with simulation. The results show that our analytical results provide accurate performance estimations for the proposed mechanism.

V. Project Self-Evaluation

In this report, we presented our research results on mobile ad hoc networks. Our contributions here include providing two efficient solutions to fair scheduling with QoS support and evaluate their performances via simulations. This proposed scheduling algorithm can be implemented at APs to achieve per-application QoS guarantee. In the next year, we will first work on the design issues and challenges of fair scheduling with power control. Afterward, we will design and evaluate our scheduling mechanism.

VI. References

[1] Haiyun Luo and Songwu Lu, “A topology-independent fair queueing model in ad hoc wireless networks,” *Proc. IEEE ICNP’00*, pp. 325-335.

[2] H. Luo and S. Lu, “A self-coordinating approach to distributed fair queueing in ad hoc wireless networks,” *Proc. IEEE INFOCOM’01*, pp. 1370-1379.

[3] H. L. Chao, and W. Liao, “Fair scheduling with QoS support in ad hoc networks,” *Proc. IEEE LCN,* 2002.

[4] Y. Cao and V. O. K. Li, ”Scheduling algorithms in broadband wireless networks,” *Proc. of the IEEE*, vol. 89, no. 1, Jan. 2001

[5] T. S. Eugene Ng, I. Stoica and H. Zhang, “Packet fair queueing algorithm for wireless networks with location-dependent errors,” *IEEE INFOCOM*, pp. 1130-1111, 1998.

[6] S. Lu, V. Bharghavan, and R. Srikant, “Fair scheduling in wireless packet networks,” *IEEE/ACM Trans. on Networking*, Vol. 7, No. 4, Aug 1999.

[7] S. Lee, K. Kim, and A. Ahmad, “Channel error and handoff compensation scheme for fair queueing algorithms in wireless networks,” *IEEE ICC*, pp. 3128-3132, 2002.

[8] Y. Cao and V. O. K. Li, “Wireless packet scheduling for two-state link models,” *Proceedings of IEEE Globecom*, Taipei, Taiwan Nov. 2002.

[9] P. Goyal, H. M. Vin, and H. Cheng, “Start-time fair queueing: a scheduling algorithm for integrated services packet switching networks,” *IEEE/ACM Trans. on Networking*, vol. 5, no. 5, pp. 690-704, 1997.

[11] E. O. Elliot, “Estimates of error rates for codes on burst-noise channels,” *Bell Systems Technical Journal* 42 (Sep. 1963) 1977-1997.

[12] H. L. Chao and W. Liao, “Credit-based slot allocation for multimedia mobile ad hoc networks,” *IEEE J. Select. Areas Commun.*, Vol. 21, No. 10, Dec. 2003, pp.1642-1651.