Available online at www.sciencedirect.com

ScienceDirect

Computer Communications 31 (2008) 1727–1738

computer communications

www.elsevier.com/locate/comcom

Performance comparison of scheduling algorithms in network mobility environment

Yaning Wang *, Linghang Fan, Dan He, Rahim Tafazolli

Centre for Communication System Research, University of Surrey, Guildford, UK

Received 22 February 2007; received in revised form 6 November 2007; accepted 6 November 2007
Available online 4 December 2007

Abstract

Network mobility (NEMO) supports a network moving as a whole, and this may cause the bandwidth on its wireless link varying with time and locations. The quick and frequent bandwidth fluctuation makes the resource reservation and admission control lack of scalability but with heavy overhead. A feasible solution for this problem is using scheduling algorithms to optimise the resource distribution based on the varying available bandwidth. In this paper, the performance comparison of several well-known priority queue (PQ) and fair queue (FQ) scheduling algorithms are given and their advantages and disadvantages in the NEMO environment are analysed. Moreover, a novel scheduling algorithm, named adaptive rotating priority queue (ARPQ), is proposed to avoid the problems of the existing algorithms. ARPQ operates with a "priority first, fairness second" policy and guarantees the delay bounds for the flows with higher priorities and maintain the reasonable throughput for the flows with lower priorities. The simulation results show that ARPQ outperforms all the existing scheduling algorithms in mobile networks, whose capacities are time-varying and location-dependent.

Keywords: Scheduling algorithms; Network mobility

1. Introduction

In recent years, network mobility (NEMO) [1,3] is received more and more attentions. Because of the high mobility of a NEMO subnet, the capacity on the wireless link between the mobile router (MR) of the NEMO subnet and the base station (BS) in the access network varies with NEMO's location change. Hence, the available radio resources can not always support all the sessions of all the users in a NEMO subnet. Although admission control and dynamic service level specification (SLS) renegotiation mechanisms [12,13] can be used to minimize the impact of the varying capacity, both of them suffer from the scalability problem caused by the frequent capacity variation. Another feasible solution for this problem is using a scheduling algorithm that can dynamically change the weights of the different traffic classes to optimize the resource distribution, and this is the focus of this paper.

In this paper, we make a comprehensive survey to the existing scheduling algorithms used in wired and wireless networks, and these scheduling algorithms can be classified as Fair Queue (FQ) and Priority Queue (PQ). The basic idea of FQ is to distribute the available resources to the different member sessions or classes with a set of pre-defined weights. Most FQ algorithms designed for the wireless network assign weights to different classes statically [23]. As improvement of static FQ algorithms, some FQ approaches apply dynamic weight distribution to deal with the varying traffic profiles or changing session requirements [15,16]. PQ algorithms schedule the different classes of flows and transmit packets with different priorities. Unlike FQ, PQ processes the packets in the queues with higher priority earlier than those with low priority. However, both FQ and PQ algorithms cannot work properly in the NEMO environment. The existing FQ algorithms, which do not consider the fluctuation of the available capacity,

* Corresponding author. Tel.: +44 1483682292.
E-mail address: yaning.wang@surrey.ac.uk (Y. Wang).

0140-3664/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.comcom.2007.11.017

can cause the *static weight-delay problem*, in which the sessions with the different priorities all suffer long delay. The PQ algorithms with static priorities emphasize the importance of the high priority queues, which allocate as much resources as possible to high priority traffics. When the resources are used up by the higher priority traffic, the lower priority traffic has to wait. This may cause the *starvation problem*, in which low priority queues cannot transfer packets at all.

In order to avoid the problems of the existing FQ and PQ algorithms in the NEMO environment, a novel scheduling algorithm, namely Adaptive Rotating Priority Queue (ARPQ) [18], is proposed in the second part of this paper. As a priority queuing algorithm, ARPQ differentiates traffic classes into different priorities. In order to keep relative fairness for the lower priority classes, ARPQ adaptively changes the weights of different priority classes with the fluctuation of the available bandwidth to leverage the use of available bandwidth, so that a maximum number of higher priority queues can be given their desired bandwidth, while a portion of bandwidth is allocated to their lower priority counterparts to share. By this means, ARPQ considers both the performance of higher priority queues and the fairness of the lower priority ones. ARPQ can not only combine the advantages of both PQ and FQ, but also take the mobile network's characteristics into account. In this way, it can guarantee the performance of higher priority queues in the tolerable delay but provide much better throughput performance to the lower priority queues in mobile networks.

The performance of the existing scheduling algorithms, e.g., several selected typical FQ and PQ algorithms and the newly proposed ARPQ, are compared in the NEMO environment via simulation. The results show that ARPQ has good tolerance to bandwidth fluctuation and guarantees queue delay for high priority queues while ensuring reasonable throughput for the low priority queues. Therefore, the starvation problem and the static weight-delay problem are effectively avoided. The overall performance of ARPQ excels those of all other comparative algorithms.

The remainder of the paper is organized as following: Section 2 analyses the problem of QoS provisioning in NEMO environment and raises the requirements of NEMO scheduling algorithms. Section 3 introduces several popular FQ and PQ algorithms and analyses their advantages and disadvantages. A novel scheduling algorithm called ARPQ is also presented. In Section 4, a simulation comparison is made to three typical scheduling algorithms and ARPQ to evaluate their performance. Section 5 makes the conclusion of the comparison.

2. Problem statements for scheduling algorithm in the NEMO environment

As the fluctuation on the available bandwidth happens on the wireless link between the MR and the BS, the uplink packets, which are transmitted from the mobile nodes (MN) in the NEMO subnet to the BS in the access network, are heavily affected by the bandwidth fluctuation. Not losing the generality, this paper focuses on the uplink scheduling in NEMO-based vehicular networks. As shown in Fig. 1(b), a simple model is used for illustration. In this model, several MNs are connected to the MR via wireless links or wired links, and they can set up sessions to the corresponding nodes in the internet. All the packets will pass the MR, and the MR then forwards them to the BS. It is noted that the MR and the BS are connected with each other via a wireless link.

Because of the high mobility of a NEMO subnet, the available bandwidth between the BS and the MR are not stable and it fluctuates in a short term, e.g. it can increase and decrease frequently and/or sharply. Theoretically, this bandwidth fluctuation problem can be solved by renegotiating the dynamic service level specification (SLS) [11–13] based on the varying bandwidth. However, such bandwidth change can happen in minutes or seconds, much more frequently than the handoffs. In such a short amount of time, dynamic SLS negotiation cannot work properly because it is too slow and will bring large overheads.

Another candidate solution is an admission control mechanism located in the MR of the mobile network, which can keep track of the available bandwidth of the mobile network and terminate some sessions if necessary in order to guarantee the normal transmission of the other sessions. However, the lack of resources caused by the bandwidth fluctuation will not continue for long, and when the bandwidth returns to normal levels, the mobile nodes with sessions terminated has to reserve the bandwidth from the MR again and restart the data transmission of the terminated sessions. The bandwidth fluctuates so frequently that the time and signalling overhead of the resource reservation will be unaffordable for the mobile network.

The third feasible solution for this problem is to develop a scheduling algorithm that can dynamically change the weights of the different traffic classes, in order to re-allocate

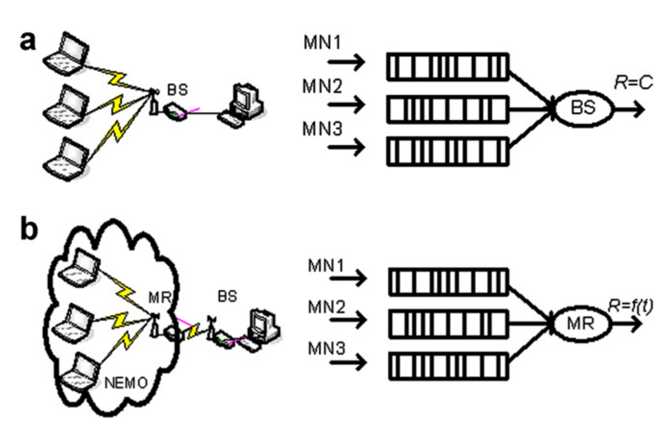

Fig. 1. Comparison of scheduling architectures of wireless networks and NEMO. (a) Legacy wireless network is connected to the access network via a wired link with constant available bandwidth; (b) NEMO is connected to the access network via a wireless link with variable available bandwidth.