Theoretical Computer Science 723 (2018) 73–83

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

# Online buffer management for transmitting packets with processing cycles

Yi-Hua Yang [a], Chung-Shou Liao [a,*], Xin Han [b], Louxin Zhang [c]

[a] Department of Industrial Engineering and Engineering Management, National Tsing Hua University, Hsinchu 30013, Taiwan
[b] School of Software, Dalian University of Technology, Dalian, China
[c] Department of Mathematics, National University of Singapore, Singapore 117543, Singapore

ARTICLE INFO

*Article history:*
Received 27 May 2017
Received in revised form 27 February 2018
Accepted 28 February 2018
Available online 5 March 2018
Communicated by T. Erlebach

*Keywords:*
Buffer management
Competitive analysis
Online scheduling
Run-to-completion

ABSTRACT

We study an online buffer management problem under the model introduced by Azar and Gilon (2015) [5] recently. Unit-sized packets arrive and are kept in a First-In-First-Out buffer of size $B$ in an online fashion at a network server. Each packet is associated with an arrival time, a value and a processing cycle time in the buffer. The *density* of a packet is defined to be the ratio of its value to processing time. It is assumed that every packet can be transmitted only after its processing cycle is completed and only the packet at the *head* of the buffer can be processed. A packet is allowed to be preempted and then discarded from the buffer. But, the value of a packet is attained only if it is successfully transmitted. Under the model, the objective of online buffer management is to maximize the total value of transmitted packets. This model finds applications to packet scheduling in communication networks.

In this study, we consider the model with constant density from a theoretical perspective. We first propose some lower bounds for the problem. Azar and Gilon obtained a 4-competitive algorithm for the online buffer management problem for packets with constant density. Here, we present a $(2 + \frac{1}{B-1})$-competitive algorithm for the case $B > 1$ as well as its generalization to the multi-buffer model. Moreover, we prove that the competitive ratio of a deterministic online algorithm is at least four when the buffer size is one. We also conduct experiments to demonstrate the superior performance of the proposed online algorithm against the previous approach.



## 1. Introduction

In a telecommunications network, modern high-speed routers involve complex interconnections and receive, buffer, process and transmit packets, where the structure of network processors (NPs) play a critical role in recent years. In this paper, we discuss the buffer management problem under the structure of NPs in a packet-switched network. Given a bounded buffer capacity, the router has to decide which packet to keep, to process and to transmit. With the increasing diversity on NPs' functions, Keslassy et al. [13] first considered heterogeneous packet processing needs, compared with the past conven-

* Corresponding author.
*E-mail addresses:* s103034515@m103.nthu.edu.tw (Y.-H. Yang), csliao@ie.nthu.edu.tw (C.-S. Liao), hanxin@dlut.edu.cn (X. Han), matzlx@nus.edu.sg (L. Zhang).

https://doi.org/10.1016/j.tcs.2018.02.035
0304-3975/

tional buffer management problems, introduced by Kesselman et al. [14] and Mansour et al. [22,23], in which the required processing time of every packet was considered to be identical, i.e. unit processing time.

We refer to Azar and Gilon's work [5] and discuss the following model of handling an FIFO (First-In-First-Out) buffer of a limited capacity $B$, where packets must be processed and transmitted in the order in which they arrive. All packets have the same size and arrive over time at a server, where each of them is associated with an arrival time, a value and a required processing cycle time. At every time slot, the server has to select some packet to process at the *head* of the buffer, and once its processing requirement is satisfied, the packet can be successfully transmitted subject to the FIFO rule and its value is obtained. Note that a packet is allowed to be preempted and then discarded from the buffer; thus, the goal is to drop packets so as to maximize the total value of packets transmitted.

In this study, we use *competitive analysis* [7,25] as a performance measure for online algorithms. Let $ALG$ be an online deterministic algorithm and $OPT$ be the optimal offline algorithm which knows every packet's arrival in advance. For an arbitrary input sequence $\sigma$, let $ALG(\sigma)$ ($OPT(\sigma)$) be the total value of all the packets transmitted by $ALG$ ($OPT$, respectively). The competitive ratio of $ALG$ is defined to be $sup_\sigma \frac{OPT(\sigma)}{ALG(\sigma)}$. This study involves discussing both lower bounds and upper bounds on competitive ratios for the problem.

**Prior work.** The buffer management problem was initialized by Kesselman et al. [14] and Mansour et al. [22,23] and it has been widely studied in the literature. There have been many different settings and variants discussed, and the best-known deterministic and randomization algorithms were presented in [3,8,9,12,21]. Recent surveys by Goldwasser [11] and Nikolenko et al. [24] provided a clear overview of the field. Moreover, for the multi-buffer model in which there are more than one buffer storing packets, Aiello et al. [1,2], Azar et al. [4] and Li [20] explored the performance of online and offline algorithms.

As mentioned, Keslassy et al. [13] first considered the buffer management problem with heterogeneous processing requirement. Later, Kogan et al. [15–19] investigated the problem with various settings. They studied the popular SRPT (shortest remaining processing time) strategy in both the push-out and non-push-out buffer management scenarios. They showed its competitive analysis as well as the corresponding lower bounds. In 2015, Azar and Gilon [5] further considered an extension in which arbitrary packet values and arbitrary required processing time are allowed. They proved that no constant-competitive algorithms exist for such buffer management problem. They also considered a model in which every packet has a constant density, i.e. a constant fraction between its value and processing time. They presented a 4-competitive algorithm, called *KeepPackets*, for solving this model. The key idea of the algorithm is as follows: it discards the packet with the smallest value in the buffer, if an arriving packet has twice larger value than the eliminated one. In this paper, we propose another strategy for solving the model with constant density, called *Run-to-Completion* (RTC), which has been used in the field of job scheduling. Here we simply introduce the concept of the RTC strategy and the details will be introduced later in Section 3.

***Run-to-Completion* (RTC) strategy:** When a packet $q$ with value $v(q)$ arrives, the algorithm discards the least valuable packet that is not at the head of the given buffer, say $p$, if $v(q) > v(p)$. That is, the RTC strategy never drops the packet when it is processed at the head of the buffer.

**Main contribution.** Our results are summarized as follows:

1. We consider the buffer management problem with constant density and propose the *Run-to-Completion* (RTC) algorithm. The competitive ratio of the algorithm can be proved within $(2 + \frac{1}{B-1})$ when $B > 1$, which improves the currently best known ratio of 4, reported in [5].
2. We present several lower bounds for different natural greedy (online) strategies for the problem. In particular, we show that no online algorithm can achieve a competitive ratio less than 4, when the buffer size $B$ is one.
3. We extend the RTC strategy to a multi-buffer model and prove that the competitive ratio stands at $(2 + \frac{1}{B-1})$. We also show the competitive analysis is tight.
4. We implement the RTC algorithm and the experimental result demonstrates that it also outperforms the *KeepPackets* algorithm [5] in practice.

## 2. Lower bounds

We discuss lower bounds on the competitive ratio of different online strategies for the buffer management problem with constant density $\rho$. Note that the following proposition holds for any online deterministic algorithms. Precisely, when the buffer is empty, the competitive ratio cannot be bounded if an online algorithm discards incoming packets.

**Proposition 1.** *Any online deterministic algorithm with a bounded competitive ratio must accept the first incoming packet, when the buffer is empty.*

First we consider non-preemptive strategies. Based on the above proposition, we show that non-preemptive strategies cannot approximate the problem well.