Preemptive Online Scheduling with Reordering

György Dósa$^1$ and Leah Epstein$^2$

$^1$ Department of Mathematics, University of Pannonia, Veszprém, Hungary
dosagy@almos.vein.hu
$^2$ Department of Mathematics, University of Haifa, 31905 Haifa, Israel
lea@math.haifa.ac.il

Abstract. We consider online preemptive scheduling of jobs, arriving one by one, on $m$ identical parallel machines. A buffer of a positive fixed size, $K$, which assists in partial reordering of the input, is available for the storage of at most $K$ unscheduled jobs. We study the effect of using a fixed sized buffer (of an arbitrary size) on the supremum competitive ratio over all numbers of machines (the overall competitive ratio), as well as the effect on the competitive ratio as a function of $m$.

We find a tight bound on the competitive ratio for any $m$. This bound is $\frac{4}{3}$ for even values of $m$ and slightly lower for odd values of $m$. We show that a buffer of size $\Theta(m)$ is sufficient to achieve this bound, but using $K = o(m)$ does not reduce the best overall competitive ratio which is known for the case without reordering, $e^{-1}$. We further consider the semi-online variant where jobs arrive sorted by non-increasing processing time requirements. In this case we show that it is possible to achieve a competitive ratio of 1. In addition, we find tight bounds as a function of both $K$ and $m$.

1 Introduction

Scheduling of jobs arriving one by one (or over list) is a basic model in online scheduling [17]. The system consists of a set of $m$ identical machines that can process a sequence of arriving jobs. Each job $j$, which has a processing time $p_j$ associated with it (also called size), needs to be assigned upon arrival. The completion time, or load, of a machine is the total time needed to process the jobs assigned to it, including idle time in which the machine is waiting for a job to be executed (if idle time exists). The goal is to minimize the maximum completion time of any machine, also known as the makespan.

We consider online and semi-online preemptive scheduling of jobs. An arriving job can be split into parts, which need to be assigned to non-overlapping time slots, possibly on different machines. Idle time is allowed, and each machine can process at most one job at each time. In the online scenario, a job must be treated before the next job is revealed. For an algorithm $A$, we denote its cost by $A$ as well. An optimal offline algorithm that knows the complete sequence of jobs in advance is denoted by $\text{opt}$. In this paper we measure the performance quality of algorithms using the (absolute) competitive ratio, which is the most common...
measure for the performance evaluation of online algorithms. The competitive ratio of $A$ is the infimum $R$ such that for any input, $A \leq R \cdot opt$.

We consider a model where a reordering buffer, of a fixed size $K > 0$, is available. This buffer can store up to $K$ unassigned jobs and thus assists in partial reordering of the input. Upon the arrival of a job, it is possible to either assign it completely to machines and time slots, or otherwise it is possible to store it in the buffer rather than assigning it. If the buffer already contains $K$ jobs, at least one of these jobs must be assigned to the machines in order to make room for the new job, or else the new job must be assigned.

Non-preemptive scheduling (i.e., the case where a job cannot be split into parts and must be processed continuously on one machine), with a reordering buffer, was previously studied in several papers [12,19,13,6,3]. The main research question in these papers was to find the effect of using a reordering buffer on the competitive ratio, that is, finding the lowest competitive ratio which can be achieved if the online algorithm is supplied with a buffer, and whether this competitive ratio is achievable only in the limit, or whether there exists a size of a buffer which allows to achieve this bound. This competitive ratio can then be compared to the best possible competitive ratio which can be achieved without a buffer. Clearly, an offline algorithm can be seen as an algorithm which uses an unbounded buffer. Limiting the online algorithm to a fixed sized buffer still means in most cases that the algorithm cannot perform as well as an optimal offline algorithm. Consequently, the competitive ratio for every value of $m$ is of interest, as well as the overall competitive ratio, which is the supremum competitive ratio over all values of $m$.

In all (non-preemptive) variants studied in the past, a finite length buffer already allows to achieve the best competitive ratio. In particular, for two identical machines, a buffer of size 1 is sufficient, as was proved by Kellerer et al. [12] and independently by Zhang [19]. For $m$ identical machines, Englert, Özmen and Westermann [6] showed that a buffer of size $O(m)$ is sufficient. For the more general case of uniformly related machines, where machines may have different speeds, it was shown [3] that for two machines, a buffer of size 2 allows to achieve the best competitive ratio. In fact, for some speed ratios between the two machines, a buffer of size 1 is sufficient, while for some other speed ratios, a buffer of size 1 provably does not allow to achieve the best bound. Note that it was shown by [6] that a buffer of size $m - 1$ reduces the competitive ratio for uniformly related machines below the lower bound of the case without reordering.

In this paper we answer analogous questions for preemptive scheduling.

Preemptive online scheduling without reordering was studied by Chen, van Vliet, and Woeginger [2] (see also [14,16,1]). They designed an algorithm of the best possible competitive ratio for any number of machines $m$. This competitive ratio is a monotonically increasing function of $m$, $\frac{m^m}{m^m - (m-1)^m}$, which implies an overall competitive ratio of $\frac{e}{e-1} \approx 1.58$. A number of papers generalized this result for uniformly related machines [18,9,7,4,10].

We study an additional variant where it is known in advance that jobs arrive sorted by size, in a non-increasing order. This common semi-online variant of