Information Processing Letters 117 (2017) 1–5

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Exact speedup factors for linear-time schedulability tests for fixed-priority preemptive and non-preemptive scheduling

Georg von der Brüggen [a], Jian-Jia Chen [a,*], Robert I. Davis [b,c], Wen-Hung Huang [a]

[a] TU Dortmund University, Germany
[b] INRIA Paris-Rocquencourt, France
[c] University of York, UK

ARTICLE INFO

Article history:
Received 14 April 2016
Received in revised form 15 July 2016
Accepted 1 August 2016
Available online 5 August 2016
Communicated by Nathan Fisher

Keywords:
Real-time systems
Speedup factors
Fixed-priority real-time scheduling
Non-preemptive and preemptive scheduling
Schedulability tests

ABSTRACT

In this paper, we investigate the quality of several linear-time schedulability tests for preemptive and non-preemptive fixed-priority scheduling of uniprocessor systems. The metric used to assess the quality of these tests is the resource augmentation bound commonly known as the processor speedup factor. The speedup factor of a schedulability test corresponds to the smallest factor by which the processing speed of a uniprocessor needs to be increased such that any task set that is feasible under an optimal preemptive (non-preemptive) work-conserving scheduling algorithm is guaranteed to be schedulable with preemptive (non-preemptive) fixed priority scheduling if this scheduling test is used, assuming an appropriate priority assignment. We show the surprising result that the exact speedup factors for Deadline Monotonic (DM) priority assignment combined with sufficient linear-time schedulability tests for implicit-, constrained-, and arbitrary-deadline task sets are the same as those obtained for optimal priority assignment policies combined with *exact* schedulability tests. Thus in terms of the speedup-factors required, there is no penalty in using DM priority assignment and simple linear schedulability tests.

1. Introduction

We consider the sporadic task model, in which a task τ_i is characterized by its worst-case execution time (WCET) C_i, its relative deadline D_i, and its period or minimum inter-arrival time T_i. The utilization U_i of task τ_i is defined as C_i/T_i.

For a task set τ, if $D_i \leq T_i$ holds for every task $\tau_i \in \tau$, the task set is said to have *constrained deadlines*. If $D_i = T_i$ holds for every task then τ is an *implicit-deadline* task set. Otherwise τ is an *arbitrary deadline* task set.

Earliest-Deadline-First Preemptive (EDF-P) scheduling is an optimal uniprocessor scheduling algorithm in the sense that if a valid schedule exists for a task set, then the schedule produced by EDF-P will also meet all deadlines [15]. In the non-preemptive case, no *work-conserving*[1] scheduling policy is optimal, since optimality can require the presence of inserted idle time. Nevertheless, EDF Non-Preemptive (EDF-NP) scheduling is optimal among all work-conserving non-preemptive scheduling algorithms [16].

In this paper we explore both fixed priority preemptive (FP-P) and fixed priority non-preemptive (FP-NP) scheduling, where each task is assigned a unique fixed-priority which is inherited by all of its jobs. Although fixed-priority scheduling policies are not optimal with respect

* Corresponding author.
E-mail address: jian-jia.chen@cs.uni-dortmund.de (J.-J. Chen).

[1] A scheduling algorithm is called work-conserving if it never idles the processor when there is a job ready to be executed.

http://dx.doi.org/10.1016/j.ipl.2016.08.001
0020-0190/

to schedulability, they have been widely adopted by both industry and academia for use in real-time systems due to their low scheduling overheads and simple implementation.

A number of different metrics can be used to quantify the quality of different scheduling algorithms and their schedulability tests. In this paper, we use the resource augmentation bound or *speedup factor* [18]. The speedup factor ρ for FP-P (FP-NP) scheduling is the minimum factor by which the processor speed needs to be increased to ensure that any task set that is schedulable by EDF-P (EDF-NP) is guaranteed (according to some schedulability test) to be schedulable using fixed priorities, assuming an appropriate priority assignment policy. As with prior work in this area, we assume that speeding up the processor by a factor of ρ implies that the WCET of each task τ_i is reduced to C_i/ρ.

We note that while previous work on speedup factors for uniprocessor systems has mainly focused on determining speedup factors assuming exact schedulability tests and priority assignment policies that are optimal in terms of schedulability, it is also interesting to explore how the required speedup factor changes with both the priority assignment policy and the schedulability tests used. In particular, is there a penalty in terms of a larger speedup factor for using a simple priority assignment policy such as Deadline Monotonic (DM) and sufficient schedulability tests that run in linear-time? Answering this question is the focus of the paper.

Contribution: We draw together results, either explicitly or only implicitly shown in previous publications [4,8,10,21,22] to build an overall picture of the exact speedup factors for linear-time schedulability tests combined with DM priority ordering. We note that since [10] and [8,22] were developed in parallel it was not possible for the authors of those papers to see the joint implications of their work until they were published. We complete this interesting picture by deriving upper bounds on the speedup factors for preemptive and non-preemptive fixed priority scheduling of arbitrary-deadline task sets assuming linear-time schedulability tests and DM priority assignment.

2. Speedup-optimal priority assignment

With Deadline-Monotonic (DM) priority assignment higher-priorities are assigned to tasks with shorter relative deadlines, with any ties broken arbitrarily to give unique priorities. DM priority assignment is what we refer to in this paper as *schedulability-optimal* for constrained-deadline task sets under FP-P scheduling [20]. A priority assignment policy P is referred to as *schedulability-optimal* with respect to a class of task sets (e.g. those with constrained-deadlines) and a fixed priority scheduling algorithm (e.g. FP-P) if all task sets in the class that are schedulable with some other priority assignment policy are also schedulable using priority assignment policy P.

DM priority assignment is not schedulability-optimal for FP-P scheduling of arbitrary-deadline task sets [19] or for FP-NP scheduling of any of the three classes of task sets, i.e., implicit-, constrained-, and arbitrary-deadline [17]. In these cases, Audsley's algorithm [1] can be used to find a schedulability-optimal priority assignment. It is also applicable to FP-P scheduling of systems in the presence of blocking [6].

We now introduce the concept of a *speedup-optimal* priority assignment for fixed priority scheduling. We refer to a priority assignment policy as *speedup-optimal* if the speedup factor that it requires when combined with an exact schedulability test is no larger than the speedup factor required by any other priority assignment policy. Again, this can be applied to different classes of task set and different fixed priority scheduling algorithms (FP-P or FP-NP). The optimality of a priority assignment policy with respect to schedulability implies that it is also speedup-optimal for the same class of task sets; however, non-optimality with respect to schedulability does not necessarily imply non-optimality with respect to the speedup factor required. This leads to the following interesting observation.

The recent work of Davis et al. [10] and von der Brüggen et al. [22] shows that DM priority assignment is a *speedup-optimal* priority assignment policy for fixed priority scheduling in *all* of its forms, i.e. FP-P and FP-NP scheduling of implicit-, constrained-, and arbitrary deadline tasks sets.

Since DM priority assignment is schedulability-optimal for implicit and constrained deadline task sets under FP-P scheduling, this implies that it is also speedup-optimal in those cases. Somewhat surprisingly, Deadline Monotonic priority assignment is also speedup-optimal for both FP-P scheduling (Theorem 1 in [10]) and FP-NP scheduling (Theorem 7 in [10]) of task sets with arbitrary deadlines. These theorems show that the exact speedup factors are unchanged when DM priority assignment is used in place of Audsley's algorithm [1]. Similarly, DM priority assignment is also speedup-optimal for task sets with implicit or constrained deadlines under FP-NP scheduling, since the upper bounds on the speedup factors proven for DM priority assignment in those cases [22] match the lower bounds determined assuming Audsley's algorithm [10,11]. This implies that the bounds are exact for both DM priority assignment and schedulability-optimal priority assignment using Audsley's algorithm.

In the remainder of the paper, we show that the upper bounds on the speedup factors for fixed priority scheduling using DM priority assignment are the same as the lower bounds (proven for exact tests and schedulability optimal priority assignment policies) even when simple linear-time schedulability tests are used. Hence we show that similar to the simplification from schedulability-optimal priority assignment to DM priority assignment, the simplification from exact pseudo-polynomial or exponential schedulability tests to simple linear-time sufficient tests also brings no additional penalty in terms of the speedup factors required.

3. Linear-time schedulability tests

We focus only on linear-time schedulability tests for fixed priority scheduling with DM priority assignment (for brevity referred to as *DM scheduling*). We assume that there are n sporadic tasks and that those tasks are indexed in order of non-decreasing relative deadlines, i.e., $D_1 \leq D_2 \leq D_3 \leq \cdots \leq D_n$. Suppose that the first $k-1$ tasks