Information Processing Letters 111 (2011) 334–337

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Comments on "Generalized rate monotonic schedulability bounds using relative period ratios"

Moonju Park

Department of Computer Science and Engineering, University of Incheon, 12-1 Songdo-dong, Yeonsu-gu, Incheon 406-772, Republic of Korea

ARTICLE INFO

Article history:
Received 19 January 2010
Received in revised form 22 December 2010
Accepted 22 December 2010
Available online 28 December 2010
Communicated by A.A. Bertossi

Keywords:
Real-time systems
Scheduling
Real-time scheduling
Rate-monotonic analysis
Schedulability test

ABSTRACT

In this Letter, it is shown that the schedulability test method for task sets with the maximum period ratio larger than or equal to 2 presented in the paper [Wei et al., Generalized rate monotonic schedulability bounds using relative period ratios, Information Processing Letters 107 (5) (2008) 142–148] is not exactly correct by presenting a counter-example. Correct sufficient conditions for using period ratios in RM schedulability test when the maximum period ratio is not less than 2 are also presented.

© 2010 Elsevier B.V. All rights reserved.

1. Introduction

From the early work of Liu and Layland [1], there has been much research for better schedulability condition on Rate Monotonic (RM) scheduling. Since the response time calculation is NP-hard [2], many researchers have studied on sufficient schedulability conditions to find a tighter schedulability bound, such as the hyperbolic bound in [3,4].

Another attempt to find a higher schedulability bound has been made using period ratios. Let the period of a task τ_i be T_i. Throughout this Letter, given a task set $\tau = \{\tau_i\}$, we assume that n tasks of a task set τ are sorted in non-decreasing order of period. Then T_1 is the shortest and T_n is the longest period. For this task set, the smallest period ratio in the set is given by $r = T_1/T_n$.

In [5], it is shown that a task set τ is schedulable by RM algorithm if the total utilization of the task set is less than or equal to

$$2r - \ln r - 1 \tag{1}$$

where $0.5 < r \leqslant 1$. When there are n tasks, the schedulability bound is

$$2r + (n-1)\left(\left(\frac{1}{r}\right)^{1/(n-1)} - 1\right) - 1. \tag{2}$$

The schedulability test using period ratio is improved in [6] using the smallest ($r_1 = T_1/T_n$) and largest ($r_2 = T_{n-1}/T_n$) period ratios. Given the smallest period ratio r_1 and the largest period ratio r_2, a task set is schedulable by RM algorithm if the total utilization is less than or equal to

$$r_1 + 1/r_2 + \ln r_2 - \ln r_1 - 2 \tag{3}$$

where $0.5 < r_1 \leqslant r_2 \leqslant 1$. For n tasks, the schedulability bound is given by

$$2r_1 + 1/r_2 + (n-2)\left((r_2/r_1)^{1/(n-2)} - 1\right) - 2. \tag{4}$$

Recently, this result is further generalized in [7]. They introduced the concept of conditional bound $CB(Z_x)$, where Z_x is a tuple of x period ratios. Given a Z_x, $CB(Z_x)$ is the lowest total utilization of all saturated task sets with period ratios of Z_x. Then any task set with Z_x that has a total utilization lower than $CB(Z_x)$ is RM schedulable. When we let $r_i = T_i/T_n$, $Z_x = (r_1, r_{n-x+1}, r_{n-x+2}, \ldots, r_{n-1})$ for $1 \leqslant x \leqslant n-1$. In [7], it is shown that

E-mail address: mpark@incheon.ac.kr.

0020-0190/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.ipl.2010.12.015

$$CB(Z_x) = 2r_1 + \sum_{i=2}^{x}(r_{n-i+2}/r_{n-i+1}) + (n-x)\big((r_{n-x+1}/r_1)^{1/(n-x)} - 1\big) - x \qquad (5)$$

for tasks with period ratio less than 2; that is, $0.5 < r_i \leqslant 1$ (or, $2T_1 < T_n$).

Though they obtain the $CB(Z_x)$ for tasks with period ratio less than 2, it is not known how we can calculate $CB(Z_x)$ when period ratio greater than or equal to 2. It was claimed in [7] that the schedulability conditions given in Eq. (5) can be applied by applying the concept of virtual period for task sets with period ratio greater than or equal to 2 (that is, $T_n \geqslant 2T_1$). However, the claim made in [7] is not exactly correct. In this Letter, a counter-example is presented showing the above claim is not applicable to all task sets. And correct sufficient conditions for using $CB(Z_x)$ in RM schedulability test are also presented.

2. Counter-example

In [7], a virtual period v_i of a task i is defined by $v_i = \lfloor T_n/T_i \rfloor T_i$. The virtual period is introduced to make the period ratios less than 2 virtually for task sets with maximum period ratio greater than or equal to 2. They claim that:

Theorem 3 of [7]. *Let z_i be the ratio of the ith smallest virtual period to T_n for $1 \leqslant i \leqslant n-1$. Then the task set τ is RM schedulable if $U(\tau) \leqslant CB(Z_x) = 2z_1 + (1/z_{n-1} + z_{n-1}/z_{n-2} + \cdots + z_{n-z+2}/z_{n-x+1}) + (\ln z_{n-x+1} - \ln z_1) - x$, where $1 \leqslant x \leqslant n-1$, $Z_x = (z_1, z_{n-x+1}, z_{n-x+2}, \ldots, z_{n-1})$, and $U(\tau)$ is the total utilization of τ.*

The above claim is not exactly correct. In the proof of Theorem 3 in [7], the authors implicitly assume that if a task set is not schedulable, then the task with the longest period, which has the lowest priority, would miss its deadline. It is not always correct, especially when the maximum period ratio is very large.

Now we present a counter-example of the above claim. Let us consider a task set $\tau = \{\tau_1 = (28, 5), \tau_2 = (57, 13), \tau_3 = (71, 17), \tau_4 = (90, 12), \tau_5 = (99, 5), \tau_6 = (1000, 1)\}$, where τ_i is associated with a tuple (T_i, C_i) where T_i is the period and C_i is the worst case execution time. The utilization of a task τ_i is given by $U(\tau_i) = C_i/T_i$, and the total utilization of a task set τ is $U(\tau) = \sum_{i=1}^{n} U(\tau_i)$. The total utilization of this task set is 0.8309 approximately. When τ is scheduled by RM algorithm, τ is not schedulable since τ_5's worst case response time is $109 > 99$, while other tasks will meet their deadlines. Calculating the virtual periods of tasks, we have:

$$v_1 = \left\lfloor \frac{1000}{28} \right\rfloor 28 = 980,$$

$$v_2 = \left\lfloor \frac{1000}{57} \right\rfloor 57 = 969,$$

$$v_3 = \left\lfloor \frac{1000}{71} \right\rfloor 71 = 994,$$

$$v_4 = \left\lfloor \frac{1000}{90} \right\rfloor 90 = 990,$$

$$v_5 = \left\lfloor \frac{1000}{99} \right\rfloor 99 = 990.$$

Then we get $z_1 = 0.969$, $z_2 = 0.980$, $z_3 = 0.990$, $z_4 = 0.990$, and $z_5 = 0.994$. Applying these values to Eq. (5), we have $U(\tau) = 0.8309 < CB(Z_5) = 2z_1 + (1/z_5 + z_5/z_4 + z_4/z_3 + z_3/z_2) + (\ln z_2 - \ln z_1) - 5 \approx 0.9696$. Even $CB(Z_1)$ is about 0.9695, so the task set passes the schedulability test based on the virtual period ratios presented in [7], although it is not schedulable.

3. Schedulability conditions for tasks with $T_n/T_1 \geqslant 2$

When $2T_1 \leqslant T_n$, Theorem 3 of [7] is only applicable to task sets whose tasks are all schedulable, or whose lowest priority task is not schedulable. Note that the lowest priority task has the longest period.

The following lemma due to Lehoczky et al. [8] gives the necessary and sufficient schedulability condition for RM algorithm.

Lemma 1. *(From [9].) Given a set of real-time tasks $\tau_1, \tau_2, \ldots, \tau_n$. Assume the tasks are ordered with increasing period, $T_1 \leqslant \cdots \leqslant T_n$. Then, a task τ_k always meets its deadline T_k under RM scheduling, if and only if there exists a time $0 < t < T_k$ such that*

$$t \leqslant \sum_{i=1}^{k} \left\lceil \frac{t}{T_i} \right\rceil C_i. \qquad (6)$$

Based on Lemma 1, the following lemma shows that the non-schedulability of the lowest priority task is not affected by replacing a higher priority task $\tau_i = (T_i, C_i)$ with another task $(x_i T_i, x_i C_i)$, where x_i is an integer and $1 \leqslant x_i \leqslant \frac{T_n}{T_i}$.

Lemma 2. *For a task set τ, if the lowest priority task τ_n is not schedulable by RM algorithm, then the task set $\tau' = \{\tau'_i = (x_i T_i, x_i C_i)\}$ for $1 \leqslant i \leqslant n$ is not schedulable with RM algorithm where x_i is an integer such that $1 \leqslant x_i \leqslant \frac{T_n}{T_i}$.*

Proof. Note that $x_n = 1$. By Lemma 1, $\sum_{i=1}^{n} \lceil \frac{t}{T_i} \rceil C_i > t$ for all $t \leqslant T_n$. Since $\lceil \frac{t}{x_i T_i} \rceil x_i C_i \geqslant \lceil \frac{t}{T_i} \rceil C_i$ for an integer $x_i \geqslant 1$ and $x_i T_i \leqslant T_n$ for $\forall i$, we have $\sum_{i=1}^{n} \lceil \frac{t}{x_i T_i} \rceil x_i C_i > t$ for all $t \leqslant T_n$. $\square$

We can see that the virtual period of [7], $v_i = \lfloor \frac{T_n}{T_i} \rfloor T_i$, is the largest x_i value for τ_i in the above lemma. Lemma 2 implies that we can determine the schedulability of a task τ with RM by testing the schedulability of each τ_i with the virtual period ratios for every task subset $\{\tau_1, \ldots, \tau_i\}$ where $1 \leqslant i \leqslant n$. This method requires sorting virtual periods for each task subset, thus has $O(n^2 \log_2 n)$ time complexity.

Another schedulability test that can be applied to task sets with $T_n/T_1 \geqslant 2$ is presented in [5].

Download English Version:

https://daneshyari.com/en/article/428647

Download Persian Version:

https://daneshyari.com/article/428647

Daneshyari.com