Resource Augmentation Bounds of EDF for Sporadic Tasks with Constrained Deadlines

Zhishan Guo
Department of Computer Science
Missouri University of Science and Technology, USA
Email: guozh@mst.edu

Xin Han
School of Software Technology
Dalian University of Technology, China
Email: hanxin@dlut.edu.cn

Abstract

Even though earliest-deadline-first (EDF) is optimal in terms of uniprocessor schedulability, it is not easy to verify the schedulability of EDF on a uniprocessor for task sets with constrained deadlines—it has been shown that the problem is actually co-NP-hard. The most efficient way to solve this problem in polynomial time is via a partially linear approximation of the demand bound function. On one hand, such an approximation is proven to have a resource augmentation factor no greater than $2 - \frac{1}{e} \approx 1.632$, where $e$ is the Euler's number. On the other hand, concrete input instances have been provided to show that the lower bound of resource augmentation factors for uniprocessor systems under such approaches is 1.5. This paper studies such strategy and the existing proofs, and presents some insights to narrow the gap between the upper and lower bounds of EDF schedulability test with approximate demand bound functions.

I. BACKGROUND

Sporadic task sets [1] are widely considered workload model in the real-time systems community in the past several decades. A piece of code (task) $\tau_i$ is characterized by a worst-case execution time (WCET) $C_i$, a minimum inter-arrival separation length (also known as period) $T_i$, and a relative deadline $D_i$. A sporadic task may trigger releases of a number of jobs, where two consecutive such releases should arrive no shorter than the period. The scheduling window of a job is determined by its release time and absolute deadline (which is $D_i$ time units after the release time).

It has been shown in [2] that EDF is an optimal policy for scheduling a task set upon a uniprocessor in a preemptive manner; i.e., there exists a correct schedule for a given task set if and only if it is correctly scheduled under EDF. The known exact EDF schedulability test for constrained deadline sporadic tasks is via checking the sum of demand bound functions ($dbf$) of all tasks $\tau_i$, as presented in:

$$\forall t : 0 < t \leq lcm\{T_i\} : \sum_i dbf(\tau_i, t) \leq t, \quad (1)$$

where

$$dbf(\tau_i, t) = \max\{0, \left\lfloor \frac{t - D_i}{T_i} \right\rfloor + 1\} \times C_i. \quad (2)$$

Although [3] showed that the critical instance is when all tasks release their first job synchronously (at time 0, without loss of generality) and subsequent job arrivals are as rapidly as possible, this is still a pseudo-polynomial time algorithm as Condition (1) must be verified at all time points $t$ within a hyper-period.$^3$

II. PRELIMINARY RESULTS

In fact, it is not possible to derive a polynomial-time exact schedulability test unless $P = NP$ since the problem has proven to be co-NP-hard [4]. As a result, approximated polynomial schedulability tests have been derived. Specifically, a linear approximation is proposed in [5], with a special case mentioned in [6]:

$$dbf^*(\tau_i, t) = \begin{cases} 0, & \text{if } t \leq D_i; \\ \left(\frac{t-D_i}{T_i} + 1\right) C_i, & \text{otherwise}. \end{cases} \quad (3)$$

Since $dbf^*(\tau_i, t) \geq dbf(\tau_i, t)$ holds for any $t \geq 0$ and it is a two-piece linear function, as demonstrated in Figure 1, it is obvious that we could use $dbf^*$ instead of $dbf$ in (1) and achieve a sufficient only, yet polynomial time EDF schedulability test.

---

1Under a correct schedule, all jobs (released by the tasks) receive enough execution time (up to the WCETs) within their scheduling windows.

2Given a certain scheduling policy, the associated schedulability test is used to verify whether correctness can be guaranteed under all circumstances.

3In practice, it suffices to verify the instances when dbf function changes its value, yet the total number of such time points is still exponentially large.
A resource augmentation factor of $\rho$ of an algorithm $A$ guarantees that if a sporadic task set is feasible on $m$ identical processors, the schedule derived from the algorithm $A$ is correct by speeding up the original platform by a factor of $\rho$. In [6], Chen and Chakraborty proved the resource augmentation bound on uniprocessor systems is at most $2 - \frac{1}{e} \approx 1.6322$. Theorem 1 of [6] shows that the resource augmentation factor is at least 1.5.

III. INSIGHTS AND ON-GOING EFFORTS

We first introduce some techniques used in [6] and then give some new techniques/insights that may be helpful in lowering the upper bound of the resource augmentation factor.

**Normalization.** Given a set of tasks $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$ with $D_1 \leq D_2 \leq \cdots \leq D_n$, we can transform it to another set $\tau'$ with property: $dbf^*(\tau, t) = dbf^*(\tau', t)$ for all $t \geq D_n$.

$$C_i' = \left(\frac{D_n - D_i}{T_i} + 1\right) \cdot C_i,$$

$$T_i' = \left(\frac{D_n - D_i}{T_i} + 1\right) \cdot T_i,$$

$$D_i' = \left(\frac{D_n - D_i}{T_i}\right) \cdot T_i + D_i.$$

The transformation first occurred in [6], here we call it as a normalization. The following results are from [6] and are also demonstrated in Figure 1: $dbf(\tau, t) \geq dbf(\tau', t)$, $dbf^*(\tau', D_n) = dbf^*(\tau', D_n)$ and $D_n' < D_i' + T_i'$ for each $\tau_i'$.

**Fixing Deadlines.** Previous work only considered the set of $\tau = \{\tau_1, \tau_2, \ldots, \tau_n\}$ with $D_n < D_i + T_i$ for $1 \leq i \leq n$.

Assume that $dbf(\tau, t) \leq t$ for all $t > 0$ and $\sum_{i=1}^{n} u_i \leq 1$, where $\frac{C_i}{T_i} = u_i$. To maximize $dbf^*(\tau, D_n)$, it is proved that there is an optimal solution $\tau$ with the following property: $D_i = \sum_{j=1}^{i} C_j$.

**Analyzing.** They relaxed the constraints $D_i + T_i > D_n$ to $D_i + T_i = D_n$, then there is only one knapsack constraint: $\sum_{i=1}^{n} u_i \leq 1$, previous work solved the problem by a greedy method, i.e., $u_i$ gets its maximal value if possible, otherwise $u_i = 0$ by setting $T_i = +\infty$. Such scheme lead to an resource augmentation bound of 1.6322.

**Weakness.** Following the above analysis we found that $dbf(\tau, D_n)$ may get too much larger than $D_n$, which violates the constraint $dbf(\tau, D_n) \leq D_n$.

**Observation 1:** Assume $T_1, T_2$ are feasible on uniprocessor and each task in $T_2$ has the same execution time. Then $\lim_{T_1 \to \infty} \sup_{T_2} dbf^*(T_1, D_n) = \lim_{T_1 \to \infty} \sup_{T_2} dbf^*(T_2, D_n)$.

**Observation 2:** Assume $T$ is a set of tasks with property: for each task we have $dbf(\tau_i, t) \leq t$, where $0 \leq t \leq D_i + T_i$. Then $\sup_{T} dbf^*(T, D_n) \leq \frac{1}{e} D_n$.

**Our Result.** We recently proved that the resource augmentation factor of uniprocessor EDF schedulability with sporadic tasks is at least $\frac{14}{9}$, which is much close to the lower bound 1.5 given in [6]. We hope eventually an upper bound of 1.5 is proved following the aforementioned insights, and thus the problem can be closed.

Acknowledgment

The authors would like to thank Prof. Sanjoy Baruah from Washington University at St. Louis and Prof. Xingwu Liu at Chinese Academy of Sciences for the fruitful discussions. Work supported by NSF Award: CNS 1755965.

REFERENCES


