

基于 EDF 抢占式调度的周期任务最小响应时间分析^{*}

宾雪莲^{1,2} 杨玉海² 金士尧¹ 宾 亚²

(国防科技大学计算机学院并行与分布国家重点实验室 长沙410073)¹ (空军雷达学院 武汉430010)²

摘 要 针对采用 EDF 抢占式调度策略的系统,在假设任务对释放偏移没有特定要求的情况下,给出使得任务响应时间最小时应满足的条件,并根据该条件得到了任务最小响应时间的计算公式。该公式的一个重要应用是用于精确估计在分布式系统中后继子任务的最大释放抖动,从而精确计算出端到端任务的最大响应时间,以正确判断任务的可调度性。

关键词 实时系统,最小响应时间,释放抖动,EDF 抢占式调度

The Best-Case Response Time Analysis of EDF Preemptive Scheduled Tasks

BIN Xue-Lian^{1,2} YANG Yu-Hai² JIN Shi-Yao¹ BIN Ya²

(National Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073)¹

(Airforce Army Radar Academy, Wuhan 430010)²

Abstract The paper presents a sufficient condition for the best-case response time of EDF preemptive scheduling tasks on the assumption that the offsets of the tasks are arbitrary. Then a formula for computing the best-case response time is shown in the paper. The most important application of the solution is in the analysis of response jitter of the distributed real-time tasks. This reduction jitter implies reduced worst-case bounds for the response time, and thus allows us to determine whether a task is schedulable correctly.

Keywords Real-time system, Best-case response time, Release jitter, EDF preemptive scheduling

1 引言

在强实时系统中,必须保证所有的请求都在其截止期限之前完成,不允许错过任何一个请求的截止期限。一般采用在最坏情况下任务的响应时间对强实时系统进行可调度性分析。如果存在最大响应时间大于截止期限的任务,则认为该系统不可调度。目前,计算任务最大响应时间的方法,已研究得较成熟;计算任务最小响应时间的方法,研究得相对较少。然而计算任务最小响应时间也很重要。例如在分布式系统中,任务的最大响应时间受到其释放抖动的影响。而任务的最大释放抖动由其前驱任务的最大响应时间和最小响应时间决定。如果释放抖动估计过大,可能将本来能在截止期限内完成的任务判定为不可调度。一般假设任务最小响应时间为其最小计算时间。该假设与实际差异较大。在文[1]中指出如果能够计算出前驱子任务的最小响应时间,将能够精确估计其后继子任务的释放抖动,从而能够精确计算端到端任务的最大响应时间,进而能够准确判定系统的可调度性。

目前对于任务请求的最小响应时间的研究不是很成熟。在文[2~10]中针对固定优先级调度,给出了任务的最小响应时间的计算公式。而基于 EDF 调度的任务最小响应时间的计算方法研究得较少。在文[11]中给出了基于 EDF 抢占式调度的任务请求响应时间的计算公式。根据该公式可以计算出一个任务在超周期(所有任务周期的最小公倍数)中所有请求的响应时间,然后求出该任务的最小响应时间。采用该方法计算量大,并且只能针对任务的释放偏移为已知的情况,不适用于对释放偏移没有特定要求的任务。因此本文将针对 EDF 调度给出任务最小响应时间的计算公式。

2 基于 EDF 抢占式调度的周期任务最小响应时间分析

假设单处理机系统,采用 EDF 抢占式调度策略。任务集 $\Gamma = \{\tau_i | 1 \leq i \leq n\}$ 中所有任务都为周期性任务,任务之间相互独立且任务对释放偏移没有特定要求,即任务的第一次请求可以在任何时候到达。任务 τ_i 由(周期 T_i 、最大计算时间 $C_i^{\max}$ 、最小计算时间 $C_i^{\min}$ 、最大释放抖动 J_i 、截止期限 D_i)表示。任务 τ_i 的截止期限为任意截止期限,即 D_i 可以小于等于周期 T_i ,也可以大于 T_i 。任务 τ_i 的请求释放抖动限制在 $[0, J_i]$ 中。任务 τ_i 第 k 次请求表示为 τ_{ik} 。请求 τ_{ik} 的到达时间为 a_i ,释放时间为 r_i ,请求完成时间为 f_i ,截止期限为 d_i ,其中 $d_i = a_i + D_i$ 。

令 $R_i^{\min}$ 表示任务 τ_i 的最小响应时间, $R_i^{\max}$ 表示任务 τ_i 的最大响应时间。

定义1 若一个请求经历最大释放抖动,则称该请求的释放时间为最迟释放时间。

本文中没考虑截止期限相等的情况。只是假设若请求 τ_{jp} ($j \neq i$) 的截止期限等于请求 τ_{ik} 的截止期限,则 τ_{jp} 的优先级低于 τ_{ik} 。

引理1 要使请求 τ_{ik} 的响应时间最小,则 τ_{ik} 的到达时间 a_i 与其释放时间 r_i 相等,所有截止期限小于 d_i 且释放时间小于 a_i 的请求在 a_i 已执行完毕,并且无其它截止期限小于 d_i 的请求在 a_i 时释放。

证明: 为了使请求 τ_{ik} 的响应时间最小,则其释放时间应尽可能地早,因此 τ_{ik} 的到达时间 a_i 与其释放时间 r_i 相等时。

^{*} 本文受国家自然科学基金资助(项目标号:60073003)。宾雪莲 博士研究生,主要研究方向为实时调度技术;杨玉海 博士研究生,讲师,主要从事计算机网络技术、智能化高速存储系统等方面的研究;金士尧 博士生导师,教授,主要从事仿真、实时、并行与分布系统等方面的研究;宾 亚 讲师,主要从事并行与分布系统方面的研究。

所有截止期限小于 d_i 且释放时间小于 a_i 的请求,其优先级高于请求 τ_{ik} 。而截止期限大于等于 d_i ($j \neq i$) 的请求,其优先级低于请求 τ_{ik} 。因此若在 a_i 时截止期限小于 d_i 的请求未执行完毕或者在 a_i 时有截止期限小于 d_i 的请求释放,则在 a_i 时 CPU 先执行这些请求,只有执行完这些请求后,才可能开始执行请求 τ_{ik} 。因此为了使请求 τ_{ik} 的响应时间最小,在 a_i 时所有截止期限小于 d_i 且释放时间小于 a_i 的请求应在 a_i 已执行完毕,并且无其它截止期限小于 d_i 的请求在 a_i 时释放。

综上引理1成立。 证毕。

由 EDF 调度策略可知,截止期限大于等于 d_i 的任务 τ_j ($j \neq i$) 的请求,对 τ_{ik} 的响应时间没有任何影响,因此以下只考虑截止期限小于 d_i 的请求。

引理2 考虑一个截止期限小于 d_i 的请求,若其最迟释放时间和到达时间在时间段 (a_i, f_i) 中,则是否考虑其释放抖动对于请求 τ_{ik} 的响应时间都没有影响;若其到达时间小于 f_i 而最迟释放时间大于等于 f_i ,或到达时间小于 a_i 而最迟释放时间在时间段 (a_i, f_i) 中,则必须考虑其释放抖动对请求 τ_{ik} 的响应时间产生影响。

证明:考虑一个截止期限小于 d_i 请求 τ_{jp} ($j \neq i$)。

若 τ_{jp} 的最迟释放时间小于 f_i 即 $t_j + J_j < f_i$,且其到达时间大于 a_i ,由于 τ_{jp} 的截止期限与其释放抖动无关并且无论推迟还是提前释放 τ_{jp} 都不会减少在 (a_i, f_i) 中释放的优先级高于 τ_{ik} 的请求产生的工作负载,因此不会对请求 τ_{ik} 的响应时间产生影响。

若 τ_{jp} 的最迟释放时间大于等于 f_i 即 $t_j + J_j \geq f_i$,且其到达时间小于 f_i ,则把 τ_{jp} 的释放时间推迟到 f_i 后,将减少在 (a_i, f_i) 中释放的优先级高于 τ_{ik} 的请求产生的工作负载,从而能够减少请求 τ_{ik} 的响应时间。

若 τ_{jp} 的最迟释放时间小于 f_i 即 $t_j + J_j < f_i$,且其到达时间小于 a_i ,则把 τ_{jp} 的释放时间推迟到 a_i 后,将会增加在 (a_i, f_i) 中释放的优先级高于 τ_{ik} 的请求产生的工作负载,从而增加了请求 τ_{ik} 的响应时间。

证毕。

定理1 当请求 τ_{ik} 满足以下条件时,其响应时间最小。

- (1) 所有截止期限小于 d_i 的请求,以及请求 τ_{ik} 的到达时间 a_i 和释放时间 r_i 应满足引理1;
- (2) 在 (a_i, f_i) 中到达的截止期限小于 d_i 的请求以最迟释放时间释放;
- (3) 所有其它截止期限小于 d_i 的请求在 f_i 时以最迟释放时间同时释放;
- (4) 在 (a_i, f_i) 中释放的截止期限小于 d_i 的请求和 τ_{ik} 都以最小计算时间执行。

证明:(条件1)由引理1可知,条件(1)是使请求 τ_{ik} 响应时间最小的必要条件。

(条件2)对于截止期限小于等于周期的任务,其最大释放抖动必小于其周期。因此对于截止期限小于等于周期的任务,在 (a_i, f_i) 中到达的该任务的请求中,不存在多个截止期限小于 d_i 而最迟释放时间大于等于 f_i 的请求。对于截止期限大于周期的任务,在 (a_i, f_i) 中到达的该任务的请求中,可能存在多个截止期限小于 d_i 而最迟释放时间大于等于 f_i 的请求。因此为了避免这种情况,要求在 (a_i, f_i) 中到达的截止期限小于 d_i 的请求都以最迟释放时间释放。由引理2可知,当在 (a_i, f_i) 中到达、最迟释放时间大于等于 f_i 的请求以最迟释放时间释放时,将会减少 τ_{ik} 的响应时间。

(条件3)设请求 τ_{jp} 的最迟释放时间为 f_i ,且 τ_{jp} 的到达时间小于 f_i 。根据引理2,若 τ_{jp} 的释放时间小于 f_i ,则会增加 τ_{ik} 的响应时间;只有当以最迟释放时间释放 τ_{jp} ,才会减少 τ_{ik} 的

响应时间。

(条件4)由抢占式调度策略可知,一个请求的响应时间与其自身以及高优先级请求的计算时间有关。因此要使得 τ_{ik} 的响应时间最小, τ_{ik} 与在 (a_i, f_i) 中释放、截止期限小于 d_i 的请求都应以最小计算时间执行。 证毕。

由文[4]可知, τ_i 的最大响应时间发生在这样的忙周期中:除 τ_i 外,所有截止期限小于 d_i 的请求都以最迟释放时间在时间 t 同时释放,并且该忙周期以时间 t 为其起始时间。因此可以看出该忙周期的起始时间与 τ_i 的最小响应时间的结束时间为同一时间。

令 Φ_i 表示 τ_i 在 (a_i, f_i) 中到达的第一个请求相对于 τ_{ik} 的到达时间。

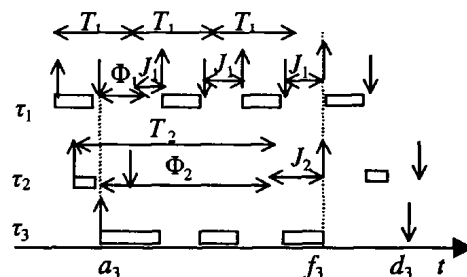


图1 任务 τ_3 在最好情况下的响应时间

图1表示当有3个任务时,任务 τ_3 在最好情况下的响应时间,其中 $\uparrow$ 表示任务请求的释放时间, $\downarrow$ 表示该请求的截止期限。任务 τ_3 请求的到达时间为 a_3 ,完成时间为 f_3 ,截止期限为 $d_3 = a_3 + T_3$ 。可以从图中看出任务 τ_1, τ_2 请求的到达时间分别为 $(\dots, f_3 - 2T_1 - J_1, f_3 - T_1 - J_1, f_3 - J_1, \dots)$, $(\dots, f_3 - J_2, \dots)$ 。任务 τ_1, τ_2 请求的截止期限分别为 $(\dots, f_3 - 2T_1 - J_1 + D_1, f_3 - T_1 - J_1 + D_1, f_3 - J_1 + D_1, \dots)$, $(\dots, f_3 - J_2 + D_2, \dots)$ 。任务 τ_1, τ_2 在 (a_i, f_i) 中的请求的释放时间分别为 $(\Phi_1 + J_1, \Phi_1 + J_1 + T_1, \Phi_1 + J_1 + 2T_1, \dots)$, $(\Phi_2 + J_2, \dots)$ 。其中 $\Phi_1 + J_1 + 2T_1 = f_3$, $\Phi_2 + J_2 = f_3$ 。在 f_3 时刻后释放的请求由于不对 τ_3 的最小忙周期产生影响,因此不需要考虑。

定理2 任务 τ_i 的最小响应时间如式(1)所示。

$$R_i^{\min} = C_i^{\min} + I_i^{\min}$$

其中, $\sum_{j \neq i} \min\{\lceil \frac{w_j^* - J_j - T_j}{T_j} \rceil_0, \lceil \frac{w_j^* - J_j}{T_j} \rceil - \lceil \frac{w_j^* - J_j + D_j - D_i}{T_j} \rceil + 1\}_0 * C_j^{\min}$

$$f(x)_0 = \max(0, f(x))$$

证明:要使得任务 τ_i 的响应时间最小,则任务 τ_i 的请求 τ_{ik} 应满足定理1。

根据抢占式调度的性质,任务的响应时间为高优先级请求对该任务的干扰时间加上其自身的计算时间,因此 $R_i^{\min} = C_i^{\min} + I_i^{\min}$ 。其中 $I_i^{\min}$ 表示高优先级请求对请求 τ_{ik} 最小干扰时间,为了使请求 τ_{ik} 的响应时间最小,应使高优先级请求的干扰时间最小。任务 τ_j 对 τ_{ik} 的干扰时间可以通过任务 τ_j 在 (a_i, f_i) 中释放的截止期限小于 d_i 的请求个数进行计算。

设 (a_i, f_i) 中释放的任务 τ_j 的截止期限小于 d_i 的请求个数为 n_j ,根据定理2可知,公式(2)成立。

$$\Phi_j + n_j T_j + J_j = f_i - a_i = R_i^{\min} \quad (2)$$

将式(2)改写为式(3):

$$n_j = \frac{R_i^{\min} - \Phi_j - J_j}{T_j} \quad (3)$$

由于 $0 < \Phi_j < T_j$,因此由式(3)可以得到不等式(4):

$$\frac{R_i^{\min} - J_j - T_j}{T_j} < n_j < \frac{R_i^{\min} - J_j}{T_j} \quad (4)$$

由于 n_i 为大于0的整数, 因此 $n_i = \left\lceil \frac{R_i^{\min} - J_i - T_i}{T_i} \right\rceil_0$ 。

n_i 的计算是基于在 (a_i, f_i) 中到达的请求的截止期限小于 d_i 的假设, 因此还必须判断这些请求的截止期限是否小于 d_i 。

设在时间段 (a_i, f_i) 中释放的任务 τ_i 的第一个请求序号为 p_i 。由定理2可知, 为了使 τ_i 的响应时间最小: 该请求的到达时间应大于 a_i 、小于 $a_i + T_i$ 。可以得到不等式(5):

$$a_i < f_i - p_i T_i - J_i < a_i + T_i \quad (5)$$

由不等式(5)推出不等式(6):

$$\frac{f_i - a_i - J_i}{T_i} - 1 = \frac{f_i - a_i - J_i - T_i}{T_i} < p_i < \frac{f_i - a_i - J_i}{T_i} \quad (6)$$

由于 p_i 为整数, 因此得到 $p_i = \left\lfloor \frac{f_i - a_i - J_i}{T_i} \right\rfloor$ 。又由于 $R_i^{\min} = f_i - a_i$, 因此 $p_i = \left\lfloor \frac{R_i^{\min} - J_i}{T_i} \right\rfloor$ 。

设在时间段 (a_i, f_i) 中释放的截止期限小于 d_i 的任务 τ_i 的请求的最大序号为 q_i 。公式(7)成立。

$$f_i - q_i T_i - J_i + D_i < d_i \quad (7)$$

其中, $d_i = a_i + D_i$ 。

由式(7)可以推出不等式(8):

$$q_i > \frac{f_i - J_i + D_i - a_i - D_i}{T_i} = \frac{R_i^{\min} - J_i + D_i - D_i}{T_i} \quad (8)$$

由于在时间段 (a_i, f_i) 中截止期限小于 d_i 的任务 τ_i 的请求个数越少, τ_i 对 τ_k 的干扰越小, 并且由于 q_i 为整数, 因此 $q_i = \left\lceil \frac{R_i^{\min} - J_i + D_i - D_i}{T_i} \right\rceil$ 。

由于 $f_i - q_i T_i$ 表示任务实例的到达时间, 因此只有当 $f_i - p_i T_i < f_i - q_i T_i$ 时, 在 (a_i, f_i) 中才存在截止期限小于 d_i 的请求。

因此, 任务 τ_i 在时间段 (a_i, f_i) 中释放的截止期限小于 d_i 的请求个数最大为 $(p_i - q_i + 1)_0$; 任务 τ_i 在 (a_i, f_i) 中释放的截止期限小于 d_i 的请求个数为: $\min(n_i, (p_i - q_i + 1)_0)$ 。

在 (a_i, f_i) 中释放的且截止期限小于 d_i 的请求对 τ_k 的最小干扰时间为:

$$I_i^{\min} = \sum_{j \neq i} \min \left\{ \left\lceil \frac{R_i^{\min} - J_j - T_j}{T_j} \right\rceil_0, \left(\left\lfloor \frac{R_i^{\min} - J_j}{T_j} \right\rfloor - \left\lfloor \frac{R_i^{\min} - J_j + D_j - D_i}{T_j} \right\rfloor + 1 \right)_0 \right\} * C_j^{\min}$$

综上定理2得证。

证毕。

由 $R_i^{\min}$ 的定义可知, $R_i^{\min} \geq C_i^{\min}$ 且 $R_i^{\min} \leq R_i^{\max}$ 。

因此 $R_i^{\min}$ 可以通过以下迭代过程计算:

$$(1) w_i^0 = R_i^{\max}$$

$$(2) w_i^{s+1} = C_i^{\min} + \sum_{j \neq i} \min \left\{ \left\lceil \frac{w_i^s - J_j - T_j}{T_j} \right\rceil_0, \left(\left\lfloor \frac{w_i^s - J_j}{T_j} \right\rfloor - \left\lfloor \frac{w_i^s - J_j + D_j - D_i}{T_j} \right\rfloor + 1 \right)_0 \right\} * C_j^{\min}$$

当 $w_i^{s+1} \leq C_i^{\min}$ 或者 $w_{s+1} = w_s$ 时, 停止迭代计算。当 $w_i^{s+1} > C_i^{\min}$ 且 $w_i^{s+1} = w_i^s$ 时, 任务 τ_i 的最小响应时间 $R_i^{\min}$ 等于 w_i^s 。

3 实例分析

本节将给出一个简单例子。先假设任务的最小响应时间为其最小计算时间, 计算出端到端任务的最大响应时间, 再采用式(1)计算出最小响应时间, 得到端到端任务的最大响应时间, 最后将两种方法得到的结果进行对比。

假设一个仅包含两个节点 N_1 和 N_2 的分布式实时系统。 τ_1, τ_2 在 N_1 上运行, τ_3, τ_4 在 N_2 上运行, 其中 τ_2 是 τ_3 的前驱子任

务, 任务之间的通信开销不计。任务的特征参数如表1所示。

表1 任务的特征参数

τ_i	$C_i^{\max}$	$C_i^{\min}$	T_i	D_i
τ_1	8	8	10	10
τ_2	4	3	30	30
τ_3	5	5	30	30
τ_4	10	10	50	50

表2为计算结果。表2(a)为假设任务的最小计算时间为其最小计算时间得到的任务的最大响应时间。表2(b)为采用公式(1)计算出的任务的最大响应时间。

表2 分析结果对比

τ_i	J_i	$R_i^{\max}$	$R_i^{\min}$	τ_i	J_i	$R_i^{\max}$	$R_i^{\min}$
τ_1	0	8	—	τ_1	0	8	—
τ_2	0	20	3	τ_2	0	20	11
τ_3	17	22	—	τ_3	9	14	—
τ_4	0	20	—	τ_4	0	15	—

(a)

(b)

从表2(a)、(b)中可以看出, 通过公式(2)计算出的最小响应时间, 使得 τ_3 的释放抖动由17降低到9, 最大响应时间由22降低到14, 任务 τ_4 的最大响应时间由20降低到15。

结论 针对 EDF 抢占式调度策略, 给出了周期任务最小响应时间应满足的条件, 并根据该条件给出周期任务最小响应时间的计算公式。最小响应时间分析的一个重要应用是用于精确计算在分布式实时系统中后继子任务的最大释放抖动, 从而能够精确估计端到端任务的最大响应时间, 正确判断任务是否可调度。因此本文所给公式对于采用 EDF 可抢占调度的分布式强实时系统的可调度性分析具有重要意义。

参考文献

- Spuri M. Holistic analysis for deadline scheduled real-time distributed systems; [Technique Report]. INRIA 1996
- Henderson W, Kendall D, Robson A. Improving the accuracy of scheduling analysis applied to distributed systems; Computing Minimal Response Times and Reducing Jitter. Real-Time Systems, 2001, 20(1): 5~25
- Redell O, Sanfridson M. Exact best-case response time analysis of fixed priority scheduled tasks. In: The Proceeding 14 th Euromicro Conference on Real-Time Systems (ECRTS'02), Vienna, Austria, 2002
- Gutierrez J C P. Best-case analysis for improving the worst-case schedulability test for distributed hard real-time systems. In: 10th Euromicro Workshop on Real Time Systems June, Berlin, Germany, 1998
- Kim T, Lee J, Chang N. Best case response time analysis for improved schedulability analysis of distributed real-time tasks. In: Proc. of ICDCS Workshops on Distributed Real-Time Systems, Taiwan, 2000
- 宾雪莲, 金士尧, 杨玉海. 基于跳跃因子模型的不规则分布蓝色任务法. 计算机研究与发展, 2003, 40(10): 1529~1534
- 杨玉海, 宾雪莲, 郑玉墙. 合作式 Web 缓存系统的性能分析. 计算机研究与发展, 2003, 40(5): 757~762
- Bin Xuelian, et al. Optimal Fixed Priority Assignment with Limited Priority Levels. Lecture Notes in Computer Science, Springer Press, 2003, 2834: 194~203
- 宾雪莲, 金士尧, 杨玉海. 周期多帧任务模型的响应时间分析. 计算机工程与科学, 2003, 25(6): 104~108
- 宾雪莲, 杨玉海, 金士尧. 基于有限优先级的静态优先级分配算法. 软件学报, 已录用, 待发表
- Devillers R, Goossens J. General Response Time Computation for the Deadline Driven Scheduling of Periodic Tasks. Fundamenta Informaticae, 1999, 40(2-3): 199~219