

# 一种新的基于优先级表的实时调度算法

王 强,徐俊刚,王宏安,戴国忠

(中国科学院软件研究所人机交互技术与智能信息处理实验室,北京 100080)

**摘 要:** 本文提出了一种新的基于优先级表的实时调度算法,称作截止期-价值密度优先(Deadline-Value Density First)算法,简称 DVDF 算法. DVDF 算法综合考虑了实时任务的截止期和价值密度两个参数,能够更好地适应不同的负载情况. 通过使用正常负载和过载情况下的典型数据对算法进行仿真研究表明,这种算法比单纯考虑截止期的 EDF(Earliest Deadline First)算法在性能方面有明显的改进,特别是在系统过载的情况下,能够优雅地降级.

**关键词:** 实时调度; 优先级; 实现价值率; 加权截止期保证率; 差分截止期保证率

**中图分类号:** TP301.6; TP316.2 **文献标识码:** A **文章编号:** 0372-2112 (2004) 02-0310-04

## A New Priority Table Based Real-Time Scheduling Algorithm

WANG Qiang, XU Jun-gang, WANG Hong-an, DAI Guo-zhong

(Intelligence Engineering Lab, Institute of Software, Chinese Academy of Sciences, Beijing 100080, China)

**Abstract:** A new priority table based real-time scheduling algorithm named DVDF (Deadline-Value Density First) is proposed. This algorithm takes the deadlines and value density of real-time tasks into account synthetically, and can adapt different load conditions better. In contrast to classical EDF (Earliest Deadline First) algorithm that only considers the deadlines of real-time tasks, DVDF algorithm is shown to improve the performance of scheduling obviously by simulation with some typical data of various parameters under normal workload and overload situation, especially under overload situation, it can degrade gracefully.

**Key words:** real-time scheduling; priority; hit value ratio; weighted guarantee ratio; differentiated guarantee ratio

## 1 引言

在当今信息时代,实时调度及其应用正越来越受到人们的广泛关注,在国民经济的各个领域,如敏捷制造、电子商务、C<sup>4</sup>I、空中交通管制、自适应容错以及机器人等,都有一定的应用,人们也对其进行了大量的卓有成效的研究. 基于优先级的实时调度算法是一种常见的调度算法,在上面提到各种实时系统中有着广泛的应用,这类算法主要包括单调速率算法 RM(Rate-Monotonic)<sup>[1]</sup>、截止期最早最优算法 EDF(Earliest Deadline First)<sup>[1,2]</sup>和空闲时间最短最优算法 LSF(Least Slack First)、价值最高最优算法 HVF(Highest Value First)<sup>[3]</sup>、价值密度最大最优算法 HVDF(Highest Value Density First)<sup>[3,4]</sup>等等. 在这些算法中,任务的优先级都是基于任务的某些特征参数,如截止期、空闲时间或关键性(即任务的重要程度或者价值)等计算而得,文献[4]中的实验结果表明了在正常负载情况下 EDF 算法表现出其最优性,而当系统过载时 HVDF 算法性能更佳. 然而,如果优先级单单根据任务的某个特征参数来确定是不够的<sup>[5,6]</sup>,如 EDF 策略将最高优先级指派给具有最早截止期的任务,LSF 策略将最高优先级指派给具有最短空闲时间的任务,但是截止期或者空闲时间短的任务不一定是

最关键的,特别是在系统过载的情况下,EDF 或者 LSF 等算法会出现急剧的性能降级,甚至导致多米诺现象<sup>[7]</sup>. 研究表明优先级驱动的调度算法应该综合考虑截止期与价值两个独立特征参数,文献[8]中就提出了一种这样的在线调度算法,称为关键性-截止期优先算法 CDF(Criticalness-Deadline First). 每个任务在到达时按照“相对截止期 ÷ 价值”分配优先级. 实验表明 CDF 算法,相对于单独使用截止期或者价值作为特征参数的算法,能够很大地改进系统的综合性能. 文献[9]中提出了基于优先级表设计与插值方法的优先级分派策略,通过综合任务的截止期与空闲时间这两个特征参数,能够有效地提高了任务调度的成功率;但是,算法要求必须事先明确特征参数的典型值并计算优先级表,而典型值的设置又会影响插值的效果.

上述算法存在的主要不足之处是:(1)由于任务的截止期与价值是两个完全不同的概念,其计量单位也不同,不能简单地将它们进行算术或者加权运算;(2)考虑系统的动态性不够,有些任务参数是动态产生的,事先确定它们的值就限制了系统的灵活性. 为了尽可能克服这些问题,本文提出一种新的基于优先级表的实时调度算法,它综合考虑了任务的截止期和价值密度两个特征参数. 实验结果表明本文所提出的算法

相对于 EDF 算法来说具有很大的性能改进,特别是在系统过载的情况下,能够优雅地降级。

## 2 DVDF 算法

在综合考虑引言中提到的各种算法优缺点的基础上,我们提出了这种综合考虑任务的截止期和价值密度的实时调度算法,称为截止期—价值密度优先 (DVDF:Deadline-Value Density First) 算法。DVDF 算法基于这样的前提:算法对任务集进行调度,不考虑任务集中任务之间的关系,使用单处理器按照优先级表来调度执行任务,属于单处理器调度算法。

### 2.1 任务描述

假设存在实时任务  $T_i$ , 用 6 元组表示为:  $T_i = a_i, C_i, d_i, D_i, V_i, vd_i$ , 其中:

(1)  $a_i$  表示任务的到达时间,即任务被启动并准备执行的时间;

(2)  $C_i$  表示任务的最坏情形执行时间,即任务在最坏情况下无中断执行所需的处理器时间;

(3)  $d_i$  表示任务的绝对截止期,即任务在这个时间应该完成执行并产生一个有价值的结果(如果没有特别说明,我们所说的截止期都是指任务的绝对截止期);

(4)  $D_i$  表示任务的相对截止期,有  $D_i = d_i - a_i$ ;

(5)  $V_i$  表示任务的价值,即任务的关键性,该任务相对于任务集中其他任务的重要程度;

(6)  $vd_i$  表示任务的价值密度,表示单位时间内任务的价值,有  $vd_i = V_i / C_i$ 。

在 DVDF 算法中,任务  $T_i$  的到达时间  $a_i$  是预先未知的,相对截止期  $D_i$  是固定的,不能满足截止期  $d_i$  的任务,价值  $V_i$  和价值密度  $vd_i$  都不再有意义。为了评估调度算法在不同负载下的性能,我们从三个方面来考察它:(1) 实现价值率 HVR (Hit Value Ratio): 调度算法调度执行任务而满足截止期所实现的累积价值与所有提交的任務价值总和的比率;(2) 加权截止期保证率 WGR (Weighted Guarantee Ratio)<sup>[6]</sup>: 指示不同关键程度任务满足截止期的情况,体现调度算法的健壮性,即在系统过载的情况下对不同关键程度任务截止期的满足程度;(3) 差分截止期保证率 DGR (Differentiated Guarantee Ratio): 用来指示不同负载情况下不同关键类别的任务的截止期保证率。

### 2.2 基于截止期与价值密度的优先级表设计

优先级分派策略可看成是一个函数,它可以针对两类不同情况:单个任务或一个任务集。当用于单个任务时,函数的结果就是对应分派策略所确定的该任务的优先级;当用于任务集时,函数的结果是任务集中任务的一个排序表,在实施调度时,优先级最高者排第一,最低者排最后。我们这里讨论的是后一种情况。

基于截止期与价值密度的优先级表设计方法的目标是在任务调度时综合考虑任务的截止期与价值密度,从而保证系统在过载的情况下能够优雅地降级,不会出现 EDF 之类调度算法中存在的多米诺现象。首先明确 DVDF 算法的调度原则:

⑧ 任务的截止期越早并且价值密度越大,则任务的优先级越高,或者在任务队列中位置越靠前;

⑨ 对于截止期与价值密度完全相同的任务,先到达者具有更高的优先级或者排在任务队列的靠前位置。

图 1 给出了一种基于截止期和价值密度的优先级表方案,其中箭头表示了任务的优先级顺序。在这种优先级分派方法中,任务的截止期序列 ( $d_1, d_2, d_3, d_4, d_5, d_6, \dots$ ) 按照升序排列,即截止期越早任务越靠前,而任务的价值密度序列 ( $vd_1, vd_2, vd_3, vd_4, vd_5, vd_6, \dots$ ) 则是采用降序排列,即任务价值密度越大越靠前。对于具有同样截止期的任务,价值密度越大,任务的优先级越高;而对于具有同样价值密度的任务,截止期越早任务的优先级越高。

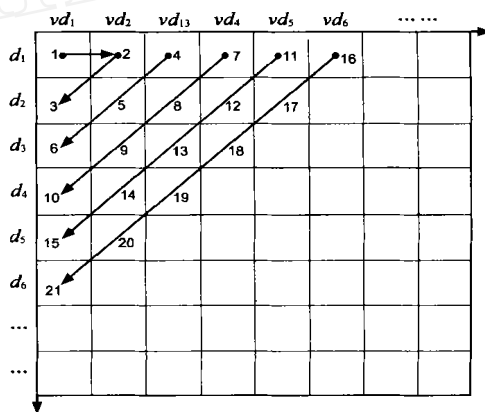


图 1 DVDF 算法的优先级表设计

在如图 1 所示的优先级表中,每个任务具有如式(1)所示的优先级:

$$p = (i + j - 1) * (i + j - 2) / 2 + i \quad (1)$$

其中  $i$  与  $j$  分别表示任务的截止期与价值密度在各自的任务序列中的位置,  $p$  值越小表示任务的优先级越高。无疑,图中每条斜线上标识的任务,其调度执行的顺序必须按照箭头所指的方向,使得任务表现为不同的优先级。即调度算法更加倾向于截止期较早的任务,价值密度相同但截止期较早的任务具有更高的优先级。

### 2.3 算法实现

在调度过程中,当新的任务到达、任务终止(完成或者夭折)时都需要调整优先级表,并且确定当前最高优先级的任务去执行。基于优先级表的特点,我们采用基于截止期的任务链表  $Q^d$  与基于价值密度的任务链表  $Q^{vd}$  交错形成一个逻辑上的优先级表,  $Q^d$  与  $Q^{vd}$  都是双向链表,便于任务结点的插入与移除。因此,向优先级表中插入一个新的任务或者移除一个终止的任务等同于分别在两个链表插入或者移除相应的任务,都能够在线性时间内完成。

## 3 实验结果和性能分析

为了评估 DVDF 算法在不同负载下的性能,我们以 EDF 算法作为基线,从实现价值率 HVR、加权截止期保证率 WGR 与差分截止期保证率 DGR 三个方面对 DVDF 算法的性能进行分析。

在我们做的所有仿真实验中,所调度的任务集由 50 个任

务  $T_i (i = 1, 2, \dots, 50)$  组成,任务的参数根据下面的方法产生:

⑧ 任务的最坏情形执行时间  $C_i$  在 5 个时间单元到 55 个时间单元之间随机选择,服从均匀分布;

⑨ 任务产生的间隔到达时间服从均值  $I_i = N \cdot C_i /$  的泊松分布,其中  $N$  表示任务集中的总任务数(此处为 50),表示期望产生的工作负载,称为标称负载;

⑩ 每个任务实例  $T_{ij}$  的松弛时间  $l_{ij}$  服从 15 到 315 之间的均匀分布;

⑪ 任务实例  $T_{ij}$  的相对截止期  $D_{ij} = C_i + l_{ij}$ ;

⑫ 设  $f$  表示任务的实际执行时间与最坏情形执行时间的比率,则  $f$  服从 0.5 到 1.0 之间的均匀分布,因此任务实例的平均执行时间为  $0.75 \cdot C_i$ ;

⑬ 为了方便按照价值密度进行统计分析,我们使得任务  $T_i$  的价值密度  $vd_i$  服从 10 到 100 之间的均匀分布,而任务价值  $V_i = vd_i \cdot C_i$ .

本文所有的实验结果都是采用 100 次独立实验结果的平均值,每次实验运行的持续时间为 300,000 个时间单元. 标称负载 是基于任务的最坏情形执行时间的,为了比较算法在不同负载情形下的性能,实验中标称负载 的取值范围为 0.5 到 3.0.

### 3.1 实现价值率 HVR

图 2 给出了 DVDF 算法和 EDF 算法在不同负载下的 HVR 变化趋势,其中横轴是标称负载,纵轴表示调度算法的实现价值率 HVR.

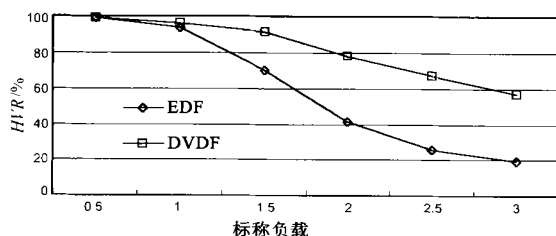


图2 EDF算法和DVDF算法的实现价值率 HVR 对比图

从图中我们不难发现,EDF算法在负载较低的情形下,表现出了很好的性能, HVR 接近 100%。但是,随着负载的不断增加,EDF算法的性能急剧下降。而 DVDF 算法在所有的负载情况下都表现出很好的性能。在负载较低的情况下,DVDF 的 HVR 与 EDF 算法相当,但是在系统过载时,DVDF 算法的性能就远远优于 EDF 算法,即使是在系统严重超载的情况 ( $= 3.0$  左右) 下,这种算法的 HVR 也在 60% 左右,这说明 DVDF 算法相对 EDF 算法而言能够更加优雅地降级。

### 3.2 加权截止期保证率 WGR

加权截止期保证率 WGR 指示不同关键程度的任务满足截止期的情况,它体现了一个调度算法的健壮性。这里,我们利用文献[6]中给出的如式(2)所示的指数加权公式来计算:

$$WGR = 100 \cdot \frac{\sum_i (i \cdot T_G^i) / \sum_i (i \cdot T_C^i)}{\sum_i (i \cdot T_G^i) / \sum_i (i \cdot T_C^i)} \quad (2)$$

式中  $i$  表示任务的关键程度,  $i = e^{i-1}$  表示不同类别任务的权重,  $T_G$  表示满足截止期的  $i$  类任务的总数,  $T_C$  表示提交的  $i$  类任务的总数。这里,我们把所有的任务按照价值密度

分为 9 类,即  $i = 1, 2, \dots, 9$ ,并对不同类别的任务的提交数量与满足截止期的任务数量进行了统计,价值密度越大,  $i$  越大。图 3 给出了 EDF 算法与 DVDF 算法在不同负载下的 WGR 变化趋势,其中横轴表示标称负载,纵轴表示不同调度算法的加权截止期保证率 WGR。

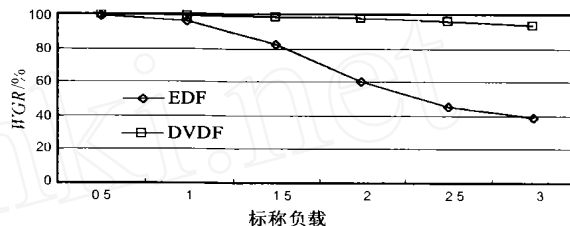


图3 EDF算法和DVDF算法的加权截止期保证率 WGR 对比

从图中我们不难发现,EDF算法随着系统负载的增加性能急剧下降,而 DVDF 算法在所有的负载情况下,都表现出很好的性能,加权截止期保证率都在 90% 以上。

### 3.3 差分截止期保证率 DGR

以实现最大的价值作为系统调度的目标,调度算法应该优先保证价值密度较大的任务满足截止期,即调度算法应该对不同类别的任务提供不同质量的服务——差分服务。这里,我们测试 DVDF 算法与 EDF 算法在不同负载下对不同类别任务(分类依据与上一节相同,也是按照价值密度把任务分为 9 类)的截止期保证率,我们称之为差分截止期保证率 DGR。

图 4(a) 与 4(b) 分别给出了 EDF 算法与 DVDF 算法在不同负载下对不同类别任务的差分截止期保证率。由图 4(a) 不难发现,EDF 算法不加区别地调度执行各种任务,因此所有任务在不同负载下具有相似的差分截止期保证率。而 DVDF 算法则执行差分服务,区别对待不同类别的任务,总是为级别高的任务提供更好的服务。从图 4(b) 能够看出,即使在系统严重过载 ( $= 3.0$ ) 的情况下,DVDF 算法仍然能够为级别最高 ( $i = 9$ ) 的任务提供接近 100% 的截止期保证率。

总而言之,EDF 算法只考虑任务的截止期,为所有任务提供相似的截止期保证率,随着负载的增加会出现急剧的性能降级。而 DVDF 算法无论是在正常负载还是过载情况下,都可以保证具有较高价值密度的任务具有较高的截止期保证率。

## 4 结论及进一步工作

本文讨论了综合考虑任务的截止期和价值密度两个特征参数的优先级设计方法,相对于先前研究中提出的算法,不需限制截止期与价值的取值范围也不需对这两种不同的特征参数进行加权运算,避免了分配优先级时的加权参数选取问题。继而,提出了基于这种优先级表的实时调度算法,并且讨论了基于多重链表的算法实现。其实,这种优先级表设计方法也适用于对其他两种不同特征参数的综合。最后,通过实验仿真,我们从实现价值率 HVR、加权截止期保证率 WGR 和差分截止期保证率 DGR 三方面对 DVDF 算法和 EDF 算法的性能进行了对比分析,结果表明 DVDF 算法相对于 EDF 算法有很大的性能改进,特别是在系统过载的情形下,算法支持优雅地降级,并且通过区分任务类别为重要任务提供等同于

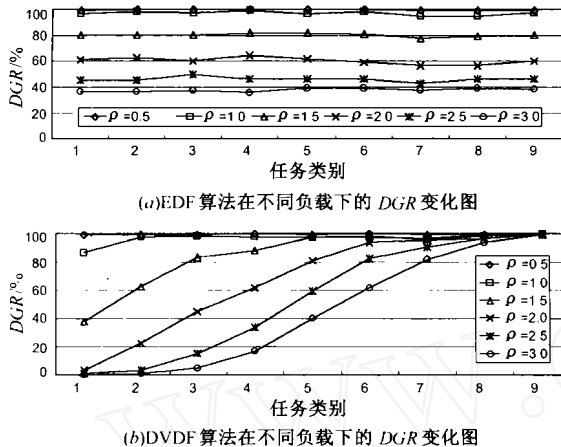


图4 EDF算法和DVDF算法的差分截止期保证率DGR对比图  
正常负载情形下的截止期保证率。

本文只是讨论和研究了综合考虑任务截止期和价值密度的单处理器调度算法,而且不支持任务间有关联关系和共享资源的情况,下一步的工作就是对这些情况进行深入研究,同时进一步研究和讨论多处理器同步和异步情况下相应算法的构造和实现问题。仿真实验中没有考虑任务抢占导致的上下文切换开销,如何在降低开销的情况下保持算法的性能优势也是我们下一步的研究工作。

#### 参考文献:

- [1] C L Liu, J W Layland. Scheduling algorithms for multiprogramming in a hard real-time environment [J]. Journal of the ACM, 1973, 20(1): 46 - 61.
- [2] R Abbott, H Garcia-Molina. Scheduling real-time transactions [J]. ACM SIGMOD Record, 1988, 17(1): 71 - 81.
- [3] E D Jensen, C D Locke, H Toduda. A time-driven scheduling model for real-time operating systems [A]. Proceedings of the 6<sup>th</sup> IEEE Real-Time Systems Symposium [C]. San Diego, California: IEEE Computer Society, 1985. 112 - 122.
- [4] G Buttazzo, M Spuri, F Sensini. Value vs. deadline scheduling in overload conditions [A]. Proceedings of the 16<sup>th</sup> IEEE Real-Time Systems Symposium [C]. Los Alamitos, California: IEEE Computer Society, 1995. 90 - 99.
- [5] A Burns, D Prasad, A Bondavalli, et al. The meaning and role of value in scheduling flexible real-time systems [J]. Journal of Systems Architecture, 2000, 46(2): 305 - 325.
- [6] S R Biryabani, J A Stankovic, K Ramamritham. The integration of deadline and criticalness in hard real-time scheduling [A]. Proceedings of the 9<sup>th</sup> IEEE Real-Time Systems Symposium [C]. Huntsville, Alabama: IEEE Computer Society, 1988. 152 - 160.
- [7] Chenyang Lu, J A Stankovic, et al. Design and evaluation of a feedback control EDF scheduling algorithm [A]. Proceedings of the 20<sup>th</sup> IEEE

Real-Time System Symposium [C]. Phoenix, Arizona: IEEE Computer Society, 1999. 55 - 66.

- [8] J Huang, J Stankovic, D Towsley, et al. Experimental evaluation of real-time transaction processing [A]. Proceedings of IEEE Real-Time Systems Symposium [C]. Santa Monica: IEEE Computer Society, 1989. 144 - 153.
- [9] 金宏, 王宏安, 王强, 戴国忠. 一种任务优先级的综合设计方法 [J]. 软件学报, 2003, 14(3): 376 - 382.

#### 作者简介:



王 强 男, 1972 年 12 月出生于山东聊城, 2000 年获山东科技大学工学硕士学位, 现为中国科学院软件研究所博士研究生, 主要研究方向为: 实时调度算法、实时数据库。



徐俊刚 男, 1972 年 10 月出生于山东聊城, 1999 年获山东大学工学硕士学位, 现为中国科学院软件研究所博士研究生, 主要研究方向为: 实时智能调度、Agent 理论及应用。



王宏安 男, 1963 年出生于江西南昌, 1999 年 6 月获中国科学院软件研究所工学博士学位, 现为中国科学院软件研究所研究员, IEEE 会员, 中国科学院软件研究所人机交互技术与智能信息处理实验室主任, 主要研究方向为: 实时智能调度、实时数据库、人机交互技术等, 在国内外期刊和会议上发表论文 20 多篇。



戴国忠 男, 1944 年出生于江苏, 现为中科院软件研究所总工程师, 主任研究员, 博士生导师, 国家 863/CIMS 领域专家, 主要研究方向为: 人机交互技术、智能信息处理、CIMS 理论及应用, 在国内外期刊和会议上发表论文 40 多篇。