

确保时态一致性的实时并发控制协议

肖迎元^{1,2}, 刘云生³

(1. 天津理工大学计算机科学与技术学院, 天津 300191; 2. 天津市智能计算及软件新技术重点实验室, 天津 300191;

3. 华中科技大学计算机科学与技术学院, 湖北武汉 430074)

摘 要: 实时控制系统通常需要维护大量的时态数据对象, 这些时态数据对象是现实世界不断变化的物理环境的逻辑表示. 应用事务通过对时态数据对象的访问来产生最终的控制活动. 在实际应用中, 实时控制系统不仅需要确保事务能在规定的期限内完成, 也需要保证事务存取的是时态一致性的数据对象. 传统的实时并发控制协议强调事务截止期的满足, 而忽略了事务对其所存取数据对象有时态一致性需求. 本文首先给出了数据和事务时态一致性的形式化定义, 在此基础上, 提出了一种新的能确保时态一致性的实时并发控制协议: 时态一致性高优先级两段锁协议 (TCHP-2PL). 为了实现更好的实时性能, 我们对 TCHP-2PL 进行了改进, 通过引入相似性概念, 提出了 TCHP-2PL 的改进版本: STCHP-2PL. 性能测试结果显示, STCHP-2PL 在确保时态一致性的同时仍能保证很好的实时性能.

关键词: 截止期; 时态一致性; 实时并发控制; 实时控制系统

中图分类号: TP311 **文献标识码:** A **文章编号:** 0372-2112 (2008) 11-2102-05

Real-Time Concurrency Control Protocol of Guaranteeing Temporal Consistency

XIAO Ying-yuan^{1,2}, LIU Yun-sheng³

(1. School of Computer Science and Technology, Tianjin University of Technology, Tianjin 300191, China;

2. Tianjin Key Laboratory of Intelligence Computing and Novel Software Technology, Tianjin 300191, China;

3. School of Computer Science and Technology, Huazhong University of Science and Technology, Wuhan, Hubei 430074, China)

Abstract: A real-time control system usually requires maintaining a large amount of temporal data objects, which are logic descriptions for the fluctuant physical environment in the real world. These temporal data objects are accessed by application transactions to generate the ultimate control actions. In the actual applications, real-time control systems require not only ensuring transactions finished in the specified time limits, but also guaranteeing temporal consistency of data objects accessed by transactions. The traditional real-time concurrency control protocols stress on meeting transaction deadlines, while ignore the requirement to temporal consistency of data objects access by transactions. This paper first gives the formal definitions of temporal consistency about data and transaction. On the basis of them, a novel real-time concurrency control protocol called TCHP-2PL (Temporal Consistency High Priority two Phase Lock) is presented. TCHP-2PL can guarantee temporal consistency. In order to obtain the better real-time performance, an enhanced TCHP-2PL called STCHP-2PL is proposed by introducing the concept of similarity. Performance tests show that STCHP-2PL can still ensure the very well real-time performance under circumstances of ensuring temporal consistency.

Key words: deadline; temporal consistency; real-time concurrency control; real-time control system

1 引言

实时控制系统 (RTCS: Real-Time Control System) 被广泛应用于具有定时限制的时间关键类应用中^[1,2], 如自动驾驶、机器导航、实时监控、过程控制等. RTCS 一方面需要维护大量的共享数据, 另一方面其应用活动 (事务) 有很强的时间性, 要求在一定的时刻或一定的时期内自外部环境采集数据、按彼此之间的联系来处理采集的数

据并做出及时的响应. 同时, 它们处理的数据往往是“短暂”的, 即只在一定的时间范围内有效, 过期则对当前决策或推导无意义.

在 RTCS 中, 事务和数据都可以具有定时限制, 系统的正确性不仅依赖于事务执行的逻辑结果, 而且依赖于逻辑结果产生的时间^[3]. RTCS 中事务的定时限制典型地表现为事务的截止期 (deadline), 它定义为事务在能给系统带来最大价值的前提下可最晚提交的时间. 事

收稿日期: 2007-10-15; 修回日期: 2008-03-28

基金项目: 国家自然科学基金 (No. 6073045); 国防预研基金 (No. 51415030203JW05); 天津市自然科学基金 (No. 08JCYBJC12400); 天津市高等学校科技发展基金 (No. 2006BA16)

务不能在规定的截止期内完成,将丧失其价值,甚至还可能带来灾难性的后果.除了持久数据对象(其值在数据对象的整个生命周期内始终有效),RTCS 还需要处理“短暂”有效的时态数据对象,时态数据对象建模外部不断变化的客观环境,是外部客观环境在计算机内的逻辑表示.时态数据对象的值通过各种传感器获得,并随着有效时间的截止而变得无效,即时态数据对象的值仅在规定的时间内(有效期)内有效.

在 RTCS 中,应用事务通过对时态数据对象的实时存取来获取外部客观环境的当前状态,进而触发相应的控制活动.当多个应用事务并发执行且都涉及到对共享时态数据对象的存取时,就存在着彼此地相互干扰,从而导致数据的不一致性.并发控制协议通过控制并发事务在读写共享数据时的执行顺序来避免并发事务间的相互干扰,从而确保共享数据的一致性.

传统的实时并发控制协议^[4~12]主要着眼于如何尽可能地减低实时事务错过截止期的比率,而忽略了实际应用中事务所存取的数据对象有时态一致性需求.在一些实际应用中,如病人监控系统中,病人的心跳、温度、血压等特征被周期性的获取并写入数据库,监控事务读取病人的心跳、温度、血压数据,并以此为依据来触发相应的活动.这里,不仅需要确保监控事务能在规定的截止期内完成,而且需要确保监控事务读取的是有效并相互一致的数据(要避免监控事务读取的是相互不一致的数据,如读取的心跳数据是第 i 次采集、写入的,而温度、血压数据则是第 $i+3$ 次采集、写入的).

2 数据与事务的时态一致性

为了便于描述,我们首先给出本文中会使用的一些符号及其含义,如表 1 所示:

表 1 本文定义的符号及其含义

符号	含义
t	系统中的任一事务
Op_i	事务对数据对象的一次读或写操作
$DS(t)$	t 需要存取的时态数据对象的集合
$DS_1(t)$	t 已存取的时态数据对象的集合
$DS_2(t)$	t 尚未存取的时态数据对象的集合
$T(t, X_i)$	t 访问时态数据对象 X_i 的时刻
$V(t, X_i)$	t 访问 X_i 时, X_i 的状态(值)
$V(Op_i, X)$	表示 Op_i 作用在数据对象 X 上的值
$P(t)$	t 的优先级

在 RTCS 中,数据和事务都有时态一致性要求.

定义 1 (时态数据对象) 时态数据对象 X 定义为一个三元组: $X = \langle V(X), ST(X), VI(X) \rangle$. 其中, $V(X)$ 表示 X 的当前状态或值; $ST(X)$ 表示采样时刻,即采样 X 所对应的外部客观环境某一特征量的时间;

$VI(X)$ 表示 X 的有效期,即自 $ST(X)$ 算起, $VI(X)$ 具有有效性的时间长度.根据上述定义,持久数据对象可看成是 $VI(X)$ 取无穷大的时态数据对象.

定义 2 (外部一致性) 时态数据对象 X 被称为满足外部一致性,如果有 $ST(X) + VI(X) \geq T_c$ 成立.这里, T_c 表示当前时刻.外部一致性确保时态数据对象的值能正确地反映外部客观环境的当前状态.当一组时态数据对象被使用时,除了外部一致性外,它们之间还存在着时间上的相互一致性问题.如时态数据对象 X 和 Y 分别表示飞行器的 x 坐标与 y 坐标,假定两个传感器分别独立地获取飞行器的 x 坐标与 y 坐标的值.当应用事务在存取 X 和 Y 时,则它必须能够保证所读取的 X 和 Y 的值是同一时刻或充分接近的两时刻的采样值.

定义 3 (相互关联集) 用来做决策或导出新数据的一组时态数据对象称为一个相互关联集.

定义 4 (相互一致性) 设 $R = \{X_1, X_2, \dots, X_m\}$ 是一个相互关联集, $V(R)$ 表示 R 中时态对象在某一时刻的状态(值)的集合,即 $V(R) = \{V(X_1), V(X_2), \dots, V(X_m)\}$. 若下列条件成立:

$$\forall V(X_i), V(X_j) \in V(R), |ST(X_i) - ST(X_j)| \leq R_{mvi}$$

则称 $V(R)$ 满足相互一致性.这里, R_{mvi} 为 R 的关联有效期,它的取值因应用语义而定,通常 R_{mvi} 取 R 中所有时态对象有效期的最小值.

定义 5 (数据的时态一致性) 若一个时态数据对象既满足外部一致性又是相互一致性的,则称它满足时态一致性.在 RTCS 中,事务除了有截止期限限制外,同样也有时态一致性需求,即要求它存取的时态数据对象必须是有效且相互一致的.

定义 6 (事务的时态一致性) 假定 $DS(t) = \{X_1, X_2, \dots, X_m\}$, 若事务 t 同时满足下列条件:

$$(1) \forall X_i \in DS(t), T(t, X_i) < ST(X_i) + VI(X_i)$$

(2) $\{V(t, X_1), V(t, X_2), \dots, V(t, X_m)\}$ 满足相互一致性则称 t 满足时态一致性.

事务的时态一致性确保事务存取的是有效且相互一致的数据.

3 并发控制协议

高优先级夭折两段锁协议(HP-2PL)是一种常用的实时并发控制策略^[6],其基本思想是:当一个高优先级(截止期紧迫)事务请求被一低优先级(截止期宽松)事务持有的数据资源而发生冲突时,为了避免“优先级颠倒”,夭折低优先级事务而让高优先级事务获得所需数据资源.该策略的优点是可避免死锁,从而消除了死锁检测与解决的开销,但该策略没有考虑到实时应用中事务时态一致性需求.为此,我们在该策略的基础上,提出了能确保事务时态一致性的实时并发控制协议(TCHP-

2PL),进一步,通过引入相似性概念,对 TCHP-2PL 进行改进,提出了 TCHP-2PL 的改进版本:STCHP-2PL.

3.1 TCHP-2PL 协议

首先,我们给出冲突事务对的定义.

定义 7 (冲突事务对) 设 t_r 和 t_h 为并发执行的一对事务,若存在 Op_i t_r , Op_j t_h 且 Op_i 和 Op_j 为一对冲突操作(即 Op_i 和 Op_j 作用在同一时态数据对象上且为读、写操作或写、读操作或写、写操作),则称 t_r 和 t_h 为一冲突事务对. 在下面的协议描述中, t_r 和 t_h 表示一冲突事务对; X_i 为一时态数据对象且 $X_i \in (DS_2(t_r) \cup DS_2(t_h))$; t_r 正在请求对 X_i 上锁而 t_h 已持有 X_i 上的锁,且 t_r 和 t_h 在 X_i 上操作为一对冲突操作.

TCHP-2PL 协议描述如下:

IF ($P(t_r) > P(t_h)$)

IF ($(DS_1(t_r) \cap \{X_i\})$ 满足相互一致性) ($T(t_r, X_i) < ST(X_i) + VI(X_i)$)

夭折 t_h , t_r 获得所申请的锁;

ELSE

t_r 被夭折;

ELSE

IF ($(DS_1(t_r) \cap \{X_i\})$ 满足相互一致性) ($T(t_r, X_i) < ST(X_i) + VI(X_i)$)

t_r 被阻塞;

ELSE

t_r 被夭折;

相比于高优先级夭折两段锁协议, TCHP-2PL 整合了时态一致性检查,因而能确保事务的时态一致性.

3.2 STCHP-2PL 协议

3.2.1 相似性概念

RTCS 中时态数据对象是外部客观环境在计算机内的逻辑表示,时态数据对象的值通过各种传感器采样获得,并被周期性地写入数据库中. 由于把采样值写入数据库就存在一定的延时,因此写入数据库中的值已经不是外部客观环境的绝对精确的反映,所以 RTCS 通常允许一定限度内的短暂的 inconsistency 存在. 传统的冲突可串行化标准对 RTCS 中的事务而言,显得过于严格,为此,我们通过引入相似性概念,来放宽对可串行化的要求.

定义 8 对于时态数据对象 X 的两个取值 $V_1(X)$ 和 $V_2(X)$, 若下面条件满足:

$$|f(V_1(X)) - f(V_2(X))|$$

则称 $V_1(X)$ 和 $V_2(X)$ 是数据相似的, 记为: $V_1(X)$

$V_2(X)$. 这里, f 表示从 X 的取值集合到实数集的一个映射; θ 为预先定义的相似阈值,其值依应用语义而定.

定义 9 设 t_m 和 t_n 为并发执行的一对事务, Op_i t_m 和 Op_j t_n 且 Op_i 和 Op_j 为作用在同一数据对象 X 上,

若下列条件满足: $V(Op_i, X) \sim V(Op_j, X)$

则称 Op_i 和 Op_j 是操作相似的, 记为: $Op_i \sim Op_j$.

定义 10 假定 SD_i, SD_j 分别表示数据库在不同时刻的两个状态, X 表示一时态数据对象, $V_i(X)$ 表示 X 在 SD_i 中的状态, $V_j(X)$ 表示 X 在 SD_j 中的状态, 若下面条件满足: $\forall X (V_i(X) \sim V_j(X))$

则称 SD_i 和 SD_j 相似, 记为: $SD_i \sim SD_j$.

定义 11 假定 Sch_a 是并发事务集 $\{t_1, t_2, \dots, t_n\}$ 的一个调度, SD_a 是 Sch_a 产生的数据库的一个状态, 若下面的条件满足: $\exists Sch_b (SD_a \sim SD_b)$

则称 Sch_a 是相似可串行化的. 这里, Sch_b 表示某一串行调度; SD_b 表示 Sch_b 产生的数据库状态.

3.2.2 STCHP-2PL 协议描述

STCHP-2PL 通过引入相似性概念来提高事务并发度. 当冲突操作为相似操作时, STCHP-2PL 并不像 TCHP-2PL 通过夭折持有锁的事务或阻塞申请锁的事务来确保并发事务的冲突可串行化, 而是允许它们并发的执行. 为此, STCHP-2PL 对锁类型进行了扩展, 在传统的 R 锁(读锁)、W 锁(写锁)基础上增加了 S 锁(相似锁), S 锁又进一步被分为 SR(相似读锁)和 SW(相似写锁). STCHP-2PL 的锁相容性矩阵如表 2 所示.

从表 2 可知, 当冲突操作为相似性操作时, 申请 S 锁不与其它任何锁冲突.

当一事务对某个时态数据对象执行读操作或写操作时, 它

表 2 锁相容性矩阵

持有 \ 请求	R	W	SR	SW
R	Y	N	Y	N
W	N	N	N	N
SR	Y	Y	Y	Y
SW	Y	Y	Y	Y

首先申请该数据对象上的 R 锁或 W 锁, 若调度器 (Scheduler) 没检测到操作冲突, 它授予该事务相应的锁; 若检测到操作冲突, 它判断冲突操作是否为相似操作, 若为相似操作, 则授予相应的相似锁.

定义 12 (冲突事务集) 假定 ST 为与事务 t_r 并发执行的事务的集合. 若 $\forall t_i \in ST$, t_r 和 t_i 都存在作用在数据对象 X 上的冲突操作, 则称 ST 为 t_r 的作用在 X 上的冲突事务集.

假定 t_r 为正在请求对 X_i 上锁的事务, Op_i 为 t_r 的作用在 X_i 上的操作; ST 为事务 t_r 的作用在 X_i 上的冲突事务集; ST_h 表示 ST 中已持有 X_i 的锁的事务集合; ST_{h1} 表示 ST_h 中作用在 X_i 上的操作与 Op_i 为相似操作的事务的集合; ST_{h2} 表示 ST_h 中作用在 X_i 上的操作与 Op_i 不为相似操作的事务的集合, 即 $ST_{h2} = ST_h - ST_{h1}$; $Max(P(ST_{h2}))$ 表示 ST_{h2} 中事务的最高优先级.

STCHP-2PL 协议可描述如下:

IF ($P(t_r) > Max(P(ST_{h2}))$)

IF ($(DS_1(t_r) \cap \{X_i\})$ 满足相互一致性) ($T(t_r, X_i) <$

$ST(X_i) + VI(X_i))$

夭折 ST_{h2} 中的所有事务, t_r 获得所需要的相似锁;

ELSE

t_r 被夭折;

ELSE

IF (($DS_1(t_r) \setminus \{X_i\}$ 满足相互一致性) $\wedge (T(t_r, X_i) < ST(X_i) + VI(X_i))$)

t_r 被阻塞;

ELSE

t_r 被夭折;

有下面的定理:

定理 1 STCHP-2PL 可确保并发事务执行满足相似可串行化。

证明 (1) 设 S 为遵循 STCHP-2PL 的并发事务集 $ST = \{t_1, t_2, \dots, t_n\}$ 的一个调度, $G = (V, E)$ 为调度 S 的优先图, 即 $V = \{t_1, t_2, \dots, t_n\}$, $E = \{(t_i, t_j) \mid t_i \text{ 和 } t_j \text{ 之间存在一对冲突操作 } Op_i \text{ 和 } Op_j \text{ 且 } t_i \text{ 执行 } Op_i \text{ 操作在 } t_j \text{ 执行 } Op_j \text{ 操作之前, } t_i \in ST, t_j \in ST\}$. 设 $G = (V, E)$ 为 G 中去掉由于相似冲突操作而导致的有向边后得到的有向图。显然根据 HP-2PL, 有向图 G 中无环。

(2) 设 e 是有向边集 $(E-E)$ 中的任一条有向边, 因为 $(E-E)$ 中的边必定是两相似冲突操作而导致的, 不妨设这两相似冲突操作分别为 Op_k 和 Op_m , 其中, Op_k 是事务 t_k 的操作, Op_m 是事务 t_m 的操作, 即 $e = (t_k, t_m)$ 。每次从 $(E-E)$ 中取出一条有向边 e , 并将其加入到有向图 G 中, 若加入后有向图 G 出现了环, 由于 Op_k 和 Op_m 为相似操作, 故交换 Op_k 和 Op_m 的执行顺序所产生的数据库状态与原执行顺序所产生的数据库状态为相似的数据库状态。故可将有向边 e 改为 (t_m, t_k) , 从而消去了图 G 中的环, 并确保了数据库状态的相似性。

(3) 重复步骤(2), 直到 $(E-E)$ 为空。

(4) 经过上面的步骤, 确保最后得到的图 G 中无环, 不妨设优先图 G 对应于事务集 $ST = \{t_1, t_2, \dots, t_n\}$ 的一个调度为 S , 显然 S 为可串行化调度。

(5) 因为调度 S 所产生的数据库状态与 S 所产生的数据库状态是相似的, 故 S 是满足相似可串行化, 即定理得到了证明。

4 性能测试与评估

由于整合了时态一致性检查, STCHP-2PL 能确保事务的时态一致性。本节通过实验来进一步评测 STCHP-2PL 的实时性能。我们在自行研发的实时数据库管理系统原型 ARTS-EDB 上完成性能测试。

我们首先比较了 STCHP-2PL、TCHP-2PL 和 HP-2PL 在实时性能方面的差异, 然后测试了不同 P_s (冲突操作为相似操作的概率) 对 STCHP-2PL 性能的影响。主要性

能指标为: 事务错过截止期的比率 (MDR), $MDR = \frac{\text{错过截止期事务的数目}}{\text{系统接纳的事务数目}}$ 。MDR 反映了系统的实时性能, 其值越小, 系统的实时性能越好。

在我们的实验中, 优先级分派采用最早截止期优先 (EDF: Earliest Deadline First) 策略。事务的截止期按如下公式计算: $Deadline = AT + SF \times ET$, 这里, AT 表示事务的到达时间; SF 表示松弛因子, 为满足均匀分布的随机变量; ET 表示事务的估计执行时间。表 3 给出了实验的主要参数, 其中, $U[i, j]$ 表示在区间 $[i, j]$ 上满足均匀分布的随机变量。图 1 显示了在 P_s 、 $RTDO$ 取缺省值 (即 $P_s = 0.5$, $RTDO = 40\%$) 时, STCHP-2PL、TCHP-2PL 和 HP-2PL 在 MDR 上的差异。当事务到达率 ($TarrRate$) 增加时, STCHP-2PL、TCHP-2PL 和 HP-2PL 的 MDR 都相应地增加, 但 STCHP-2PL 的 MDR 要明显低于 TCHP-2PL 和 HP-2PL 的 MDR。这是由于 STCHP-2PL 引入了相似性概念, 对传统的冲突可串行进行了放松, 从而提高了事务的并发度。

图 2、图 3 比较了 $RTDO$ 对 STCHP-2PL、TCHP-2PL 和 HP-2PL 实时性能的影响。当 $RTDO$ 增大时, STCHP-2PL、TCHP-2PL 和 HP-2PL 的 MDR 都相应地增加, 但对 HP-2PL 的影响最小, 这是由于 HP-2PL 没有考虑事务的时态一致性要求。尽管 $RTDO$ 对 STCHP-2PL 的影响较大, 但是由于引入了相似性概念, STCHP-2PL 的实时性能仍然明显地优于 HP-2PL 和 TCHP-2PL。

图 4 显示了在事务到达率 $TarrRate = 20 \text{ trans./second}$ 时, P_s 对 STCHP-2PL 性能的影响。如图 4 所示, 随着 P_s 的增加, STCHP-2PL 的实时性能相应地提高。这是由于随着 P_s 增加, 事务并发度也相应增加的缘故。

表 3 主要实验参数

参数	含义	取值范围	缺省值
$NTDO$	时态数据对象的数目	100 ~ 1000	500
$RTDO$	时态数据对象占全部数据对象的百分比	10% ~ 80%	40%
$TarrRate$	每秒钟事务到达数目	5 ~ 40	20
NTO	事务包含的读、写操作的数目	4 ~ 8	$U[4, 8]$
AET	单个数据操作的平均执行时间 (ms)	0.1 ~ 0.8	0.4
P_u	更新操作的概率	0.2 ~ 0.8	0.4
SF	松弛因子	2.0 ~ 6.0	$U[2, 6]$
P_s	冲突操作为相似操作的概率	0.1 ~ 0.9	0.5

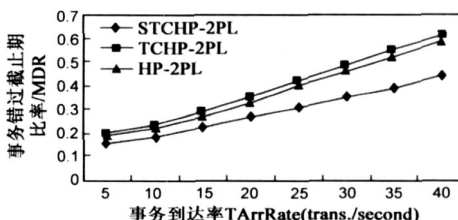


图1 三种实时并发控制协议在MDR上的比较

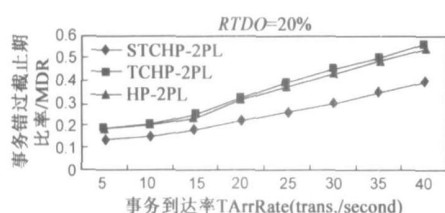


图2 RTDO=20%时三种实时并发控制协议在MDR上的比较

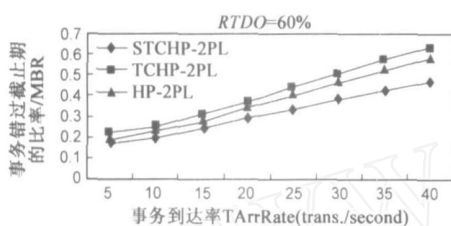
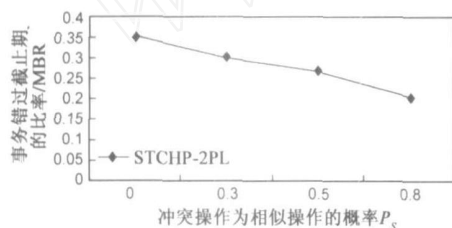


图3 RTDO=60%时三种实时并发控制协议在MDR上的比较

图4 P_s 对STCHP-2PL实时性能的影响

5 结束语

RTCS 被广泛地应用于具有定时限制的时间关键类应用中。通常,这些应用不仅需要确保事务(应用活动)能在规定的期限内完成,也要求保证事务存取的是满足时态一致性的数据。传统的实时并发控制协议强调的是如何尽可能地减低事务错过截止期的比率,而忽略了事务对其存取的数据对象有时态一致性要求。本文在给出数据和事务时态一致性定义的基础上,提出了一种新的实时并发控制协议:STCHP-2PL。STCHP-2PL 整合了时态一致性检查,因而能确保事务时态的一致性,同时,通过引入相似性概念,STCHP-2PL 对传统的冲突可串行化进行了放松,提高了事务的并发度。性能测试显示,STCHP-2PL 在确保事务时态的一致性的同时,仍能保证很好的实时性能。

参考文献:

- [1] Ramamritham K, Son S H. Real-time databases and data Services[J]. Real-Time Systems, 2004, 28(2-3): 179-215.
- [2] 谭朋柳,金海,张明虎. 用于开放式系统的二维优先级实时调度[J]. 电子学报, 2006, 34(10): 1773-1777.
Tan Peng-liu, Jin Hai, Zhang Ming-hu. Two-dimensional priority real-time scheduling for open systems[J]. Acta Electronica Sinica, 2006, 34(10): 1773-1777. (in Chinese)
- [3] Stankovic J. Real-time and embedded systems[J]. ACM Computing Surveys, 1996, 28(1): 205-208.

- [4] Sha L, Rajkumar R, Son S H. A real-time locking protocol[J]. IEEE Transactions on Computer, 1991, 40(7): 793-800.
- [5] Lam K Y, Kuo T W. Concurrency control in mobile distributed real-time database systems[J]. Information Systems, 2000, 25(4): 261-286.
- [6] Abbott R, Garcia M R. Scheduling real-time transactions: a performance evaluation[J]. ACM Transactions on Database Systems, 1992, 17(3): 513-560.
- [7] Song X, Liu J W S. Maintaining temporal consistency: pessimistic vs. optimistic concurrency control[J]. IEEE Transactions on Knowledge and Data Engineering, 1995, 7(5): 786-796.
- [8] Lau C P M, Lee V C S. Real-time concurrency control for data intensive applications[A]. Joseph K Ng. Embedded and Real-Time Computing Systems and Applications[C]. Washington, USA: IEEE Computer Society, 2005, 337-442.
- [9] Alejandro L I, Luis A. A relaxed temporal consistency approach for real-time concurrency control[A]. Silvia C. Electronics, Robotics and Automotive Mechanics[C]. Washington, USA: IEEE Computer Society, 2006, 274-280.
- [10] Lindström J. Relaxed correctness for firm real-time databases[A]. Embedded and Real-Time Computing Systems and Applications[C]. Washington, USA: IEEE Computer Society, 2006, 82-86.
- [11] Liao G Q, Liu Y S, Wang L N, et al. Concurrency control of real-time transactions with disconnections in mobile computing environment[A]. Computer Networks and mobile Computing[C]. Washington, USA: IEEE Computer Society, 2003, 205-212.
- [12] Kang K D, Son S H, Stankovic J A. Managing deadline miss ratio and sensor data freshness in real-time databases[J]. IEEE Transaction on Knowledge and Data Engineering, 2004, 16(10): 1200-1216.

作者简介:



肖迎元 男, 1969年12月生于湖南邵阳市。博士, 副教授。IEEE会员, 中国计算机学会高级会员。主要研究方向为实时信息处理、移动计算、现代数据库理论与集成实现。
E-mail: xyacat@tom.com



刘云生 男, 1940年10月生于湖南衡阳。教授, 博士生导师。主要研究方向为实时信息处理、现代数据库理论与集成实现。