

嵌入式控制驱动状态转移事件的预测调度模型

张 晶, 陈 垚, 孙 俊, 范洪博

(昆明理工大学 信息工程与自动化学院, 昆明 650500)

摘 要: 在嵌入式控制系统执行状态转移任务的过程中, 存在事件可调度性难以保证和抢占任务成本高的问题。为此, 构建一种结合前馈预测模型与反馈监测模型的预测调度模型。增加同时具有价值量约束与时钟约束的实时约束标签, 以实时调整事件在释放队列中的顺序, 提高控制系统的实时性与确定性。仿真结果表明, 该模型能有效提高控制系统性能, 稳定系统作用时间。

关键词: 嵌入式控制; 状态转移; 调度; 确定性; 约束条件

中文引用格式: 张 晶, 陈 垚, 孙 俊, 等. 嵌入式控制驱动状态转移事件的预测调度模型[J]. 计算机工程, 2017, 43(8): 90-94, 100.

英文引用格式: Zhang Jing, Chen Yao, Sun Jun, et al. Predictive Scheduling Model of Embedded Control Driving State Transition Events[J]. Computer Engineering, 2017, 43(8): 90-94, 100.

Predictive Scheduling Model of Embedded Control Driving State Transition Events

ZHANG Jing, CHEN Yao, SUN Jun, FAN Hongbo

(Faculty of Information Engineering and Automation, Kunming University of Science and Technology, Kunming 650500, China)

[Abstract] In the execution process of status transfer tasks for embedded control systems, the schedulability of events is difficult to guarantee and it needs high costs when a task is preempted. Aiming at these problems, this paper constructs a predictive scheduling model combining Feedforward prediction Model (FFM) and Feedback Monitoring Model (FMM). It adds the real-time label of constraint condition including value and timing constraints, so as to adjust the order of events in the released queue in real time and improve the real-time performance and determinacy of the control system. The result of simulation shows that the proposed model can improve the performance of control system and stabilize system's effect time.

[Key words] embedded control; state transition; scheduling; determinacy; constraint condition

DOI: 10.3969/j.issn.1000-3428.2017.08.015

0 概述

嵌入式控制通过计算机执行预设进程控制物理实体对环境的感知与识别。基于网络的实时信号控制已被广泛用于交通、航空、医疗、工业等领域, 非实时的物理过程通过对计算机构建语义、建立系统、设计硬件描述语言、对事件进行仿真与构建通信网络^[1]扩展到实时控制过程的前提, 是在不牺牲系统性能的情况下尽可能地提高控制系统的实时性与确定性。设计具有实时性的事件驱动计算机系统状态转移模型, 需要考虑程序执行时所用能耗及时间, 重点是尽可能地提高事件驱动系统状态控制的精确度和有效性^[2-4]。

嵌入式控制事件驱动系统状态转移是计算机系统的执行过程。考虑控制事件输出为时间间隔任意

小的序列, 在执行序列中对事件添加标签, 该标签包括约束系统实时性的时间标签和提高控制系统确定性的价值量标签^[5]。加入时钟约束与价值量约束条件, 相当于在早期研究的时间信息系统, 如实时自动机^[6]、时序图方法^[7]、时间 Petri 网^[8]的状态转移中引入约束条件, 其中时钟约束条件类似于截止期的上限时间与下限时间, 价值量约束类似于数据质量 (Quality of Data, QoD) 的最佳有效值。

本文基于嵌入式控制驱动状态转移事件^[2]的约束条件, 构建一种驱动状态转移事件序列的预测调度模型 (Feedforward and Feedback-Predictive Scheduling Model, FF-PSM)。模型中事件的释放队列与状态转移过程要求顺序一致, 程序唯一运行^[9-10]。由于反馈控制器 (Feedback Controller, FC) 只连续监测系统性能误差, 若有极端情况, 可能导致崩溃, 因此引入前馈

基金项目: 国家自然科学基金 (61562051); 云南省应用基础研究计划重点项目 (2014FA029)。

作者简介: 张 晶 (1974—), 男, 教授、博士, 主研方向为实时嵌入式软件; 陈 垚, 硕士研究生; 孙 俊 (通信作者)、范洪博, 博士。

收稿日期: 2016-06-30 **修回日期:** 2016-08-08 **E-mail:** 414356764@qq.com

控制器(Feedforward Controller, FFC)从本质上提高系统鲁棒性。基于超致密时间模型(Super Dense Time Model, SDT)^[13],在事件标签中加入时钟约束与价值量约束,得到前馈预测^[11]与反馈监测^[12]结合的FF-PSM模型,在驱动状态转移时,前馈预测控制事件不低于有效价值量阈值,反馈监测状态转移的性能,提高控制系统的实时性与确定性。

1 事件驱动状态转移过程

在嵌入式系统中,控制系统状态转移的过程为计算机执行预设进程控制物理实体执行操作任务的过程,其事件可认为是输入的外部指令、数据或变量,在实际情况下系统状态转移需要由输入事件预测下一状态,为考虑控制系统的实时性与确定性,定义事件驱动状态转移过程含有时间约束与价值量约束。

1.1 状态转移过程定义

状态转移过程为事件从释放队列获得释放时间开始,到该事件释放至相应状态以驱动状态转移的过程。

定义1 驱动状态转移的事件 a 在队列中获得释放时间 $t = t_r$ 后,事件携带含有时间约束与价值量约束标签 $a(\tau, j)$,在时刻 t_i ,事件 a_{i+1} 驱动 s_i 状态转移。

事件驱动状态转移过程如图1所示,在获得释放事件前,事件在事件队列中排队等待获得释放时间。事件获得释放时间 t_r 后即可在 $t = t_r$ 时释放至相应状态以驱动其转移。例如,事件 a_1 在取得释放时间 t_{r1} 后,在 $t = t_{r1}$ 释放至 s_0 ,此时其时钟约束为 x_1 ,价值量约束为 j_1 ;类似地,事件 a_i 驱动状态 s_{i-1} 转移,包含用以表示时间约束的时钟约束 x_i ,以及用以表示进程性能约束的价值量约束 j_i 。

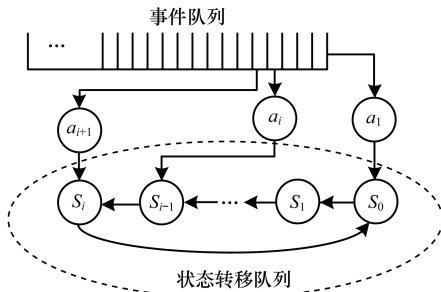


图1 事件驱动状态转移过程

1.2 事件驱动状态转移的约束条件

在事件驱动状态转移过程中加入时间约束条件和价值量约束条件以满足系统的实时性与确定性。

定义2 嵌入式控制驱动状态转移事件可描述为: $(s_0, v_0)(a_1, \tau_1, j_1) \rightarrow (s_1, v_1)(a_2, \tau_2, j_2) \rightarrow \dots \rightarrow (s_n, v_n)(a_{n+1}, \tau_{n+1}, j_{n+1})$,其中,驱动状态 $s \in S(T, V) = T \rightarrow V, t \in T, v \in V$;事件驱动状态转移的约束条件 $a \in a(\tau, j)$

$= (a, (\tau, n), (\tau, t_r, v)), v \in j \in J, \tau \in \delta$ 。

定义3 令约束条件函数 $F = f(\tau, j)$,状态转移过程函数 $\inf(r), r = (s, v)$,当且仅当 $\inf(r) \cap F \neq \emptyset$ 时,状态转移元组 $\langle A, S, s_0, J, C, E, F \rangle$ 在 $r = (s, v)$ 能正常执行任务。其中, $E \subseteq S \times S \times A \times 2^C \times \Delta(C)$ 为事件驱动状态转移路径; C 表示二进制时钟集; $\delta \subseteq \Delta(C)$ 表示时间约束; $\lambda \subseteq C$ 表示在转移过程中重设时钟 x 。

由定义2与定义3可得:在事件驱动状态转移过程中,要激发事件驱动状态执行转移任务,需要以满足约束条件为前提。

定义4 设时钟约束函数 $\Delta(X)$ 和价值量约束函数 $\Delta(J)$,由定义1、定义3可得: $x \in X, j \in J, v$ 为函数 $\Delta(J)$ 变量,常数 $c \in Q, Q$ 为非负有理数。

$$\begin{cases} \Delta(X) : = x \leq c \mid x \geq c \mid \neg \delta(x) \mid \delta_1(x) \wedge \delta_2(x) \\ \Delta(J) = v \leq c \mid v \geq c \mid \neg j(v) \mid j_1(v) \wedge j_2(v) \end{cases}$$

(1)

当事件 a_i 到达状态 s_{i-1} 时,当满足式(1)中 $\Delta(X) = \Delta(J) = \text{ture}$ 时,则 (s, v) 满足状态转移条件。

定义4可描述事件驱动状态转移过程中的约束条件,如图2所示,事件 a_i 驱动状态 s_{i-1} 转移至状态 s_i ,事件 a_{i+1} 驱动状态 s_i 转移至 s_{i-1} 。当事件 a_i 携带含有约束条件标签 (τ_i, j_i) 执行驱动任务时,可在状态转移过程中重设时钟 $x_i := \lambda$,此时,价值量的有效值 $v_i = v_i^{\max}$,显然满足定义4及式(1)要求,可执行状态转移任务。当事件 a_{i+1} 准备驱动状态 s_i 执行转移任务时,由约束条件标签 (τ_{i+1}, j_{i+1}) 执行驱动任务,考察时钟约束条件是否满足可执行条件 $c_{\delta i+1}$,即 $\{(x_{i+1} < c_{\delta i+1})?\}$,同时测试价值量有效值是否为可正常运行驱动的有效值 $c_{v i+1}$,即 $\{(v_{i+1} \geq c_{v i+1})?\}$ 。

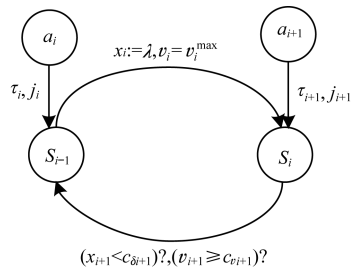


图2 状态转移约束条件

1.3 价值量有效值特性

事件在 $t = t_r$ 时从队列中释放,在到达状态激发其发生转移的过程中,事件价值量随时间 t 发生变化,对触发状态转移性能随之变化。

定义5 任意状态 (s_i, v_i) 得到事件 a_{i-1} 后,其事件驱动状态转移有效值满足 $v_i(t) + t_r \geq t$,即转移时间 t 由释放时间 t_r 与作用时间 $v_i(t)$ 构成。

令 $v_i(t) = v_i(t+1) - \Delta v_i(t)$,则事件有效值变化函数为:

$$K_i(t) = \{(t_r + v_i(t)) - t, t \leq t_r + v_i(t)\} \quad (2)$$

价值量随有效值变化函数为:

$$J_i(t) = 1 - \frac{K_i(t)}{\nu_i^{\max} - \nu_i^{\min}} \quad (3)$$

由式(2)、式(3)可得事件 a_i 驱动状态转移所具有的价值量有效值 ν_i 在收敛区间 $(\alpha_i, \beta_i) \subseteq (\tau_i - \tau_{i-1})$ 上有最好有效性 $\nu_i^{\max}$ 使 $J_i(t) = 1$ 和最差有效性 $\lim_{t \rightarrow \infty} \nu_i^{\min}$, 使 $J_i(t) = 0$, 在区间 $(\nu_i^{\max}, \nu_i^{\min})$ 上 $J_i(t)$ 单调递减。

1.4 事件驱动状态转移在 SDT 模型下的确定性

定义 6 设超致密模型 $t = (\tau, n)$, $\tau \in \mathbb{Q}$, $n \in \mathbb{N}$, 事件驱动状态转移的价值量约束 $j = (\tau, t_r, \nu)$, $\tau \in \mathbb{Q}$, $t_r \subseteq t$, $\nu \subseteq \tau$. 在超致密模型下, 有每一个时钟记录 $x \in C$ 满足状态转移约束条件的实时状态转移约束 n_x 和 ν_i , 符合在状态转移过程中有单一的初始状态 s_0 、确定的输入事件 a_i 以及唯一确定的下一状态 s_i , 即状态转移元组 $\langle A, S, s_0, J, C, E, F \rangle$ 是唯一确定的。

如图 1 所示, 状态转移队列 $s_0 \rightarrow s_1 \rightarrow \dots \rightarrow s_i \rightarrow s_0$ 是唯一确定的, 队列由初始状态 (s_0, ν_0) 开始, 每一个状态 (s_i, ν_i) 由其确定的输入事件 a_{i+1} 驱动转移, 并且所转移的下一状态也是唯一确定的 (s_{i+1}, ν_{i+1}) 。状态转移过程是唯一确定的, 而事件释放至触发驱动则与释放时间及驱动状态转移性能有关。

2 预测调度模型

通过前馈预测与反馈控制相结合的预测调度模型, 对事件队列的释放事件进行干预, 调整释放时间, 以获得最合适的状态转移性能, 提高系统鲁棒性, 控制系统最优的实时性与确定性。

2.1 模型原理

本文设计一种基于软实时应用系统的嵌入式驱动状态转移事件预测调度模型。加入前馈控制器用于前馈预测, 反馈控制器用于控制事件驱动状态转移的性能, 结合前馈预测价值量有效值的阈值与反馈控制对性能的检测, 干预事件释放时间, 对事件释放顺序进行调度^[14-15]。

由定义 1 ~ 定义 3 可知, 事件队列所释放的事件驱动状态转移队列中状态转移过程满足状态转移元组 $\langle A, S, s_0, J, C, E, F \rangle$, 并且由定义 6 可得该转移过程为唯一确定的。

如图 3 所示, 事件由驱动器传输至事件队列, 在队列中等待系统分配释放时间 t_r , 由事件队列长度及已分配的 t_r 发送至前馈控制器进行处理, 前馈控制器预测得到事件 a_i 到达状态 s_{i-1} 驱动其进行转移任务的有效值阈值 $V_f(t)$ 并得出目标性能。在事件 a_i 驱动状态 s_{i-1} 转移后, 由事件驱动状态转移的性能分析器分析转移过程的性能指标, 与前馈预测的目标性能进行比较, 得出误差 $Error$, 由反馈控制器进行误差分析,

得出调整后的有效值阈值 $V_\xi(t)$, 结合 $V_f(t)$, $V_\xi(t)$ 得到用于调度事件的有效值阈值 $V_\psi(t)$, 即:

$$V_\psi(t) = V_f(t) + V_\xi(t)$$

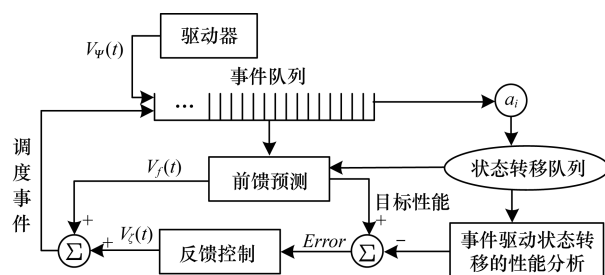


图 3 系统调度控制模型

2.2 前馈控制器设计

引入一种含有前瞻性 (look-ahead)^[16-17] 的前馈控制器 (vt-FFC) 用于预测满足状态转移性能的事件价值量有效值。由 vt-FFC 构成的前馈预测模型 (Feedforward prediction Model, FFM) 数据由事件队列及状态转移队列提供, 在 SDT 模型下, 对事件释放时刻到状态转移完成时刻这个时间段进行分析, 得出结论用于调整事件队列中事件释放时间的分配。

在事件 a_i 从驱动器输出到驱动状态 s_{i-1} 完成转移这个过程中所用的时间可分为 3 个阶段, 即在事件队列中的排队时间 q_i 、事件传输至状态并触发驱动发生的驱动延时 c_i 和状态从开始执行状态转移到完成状态转移所用的时间 e_i 。

采用 vt-FFC 的 FFM 模型调度过程如图 4 所示, 其中, c_i 表示事件的平均驱动延时; e_i 表示事件驱动状态转移过程所用的平均时间, 随着事件越来越多, 事件队列中累积的事件量也越来越多, 所以, 排队时间 q_i 随系统时间越来越长。设在系统计划完成的状态转移中需要释放 N_x 个事件, 在图 4 中, 某一事件 a_i 的释放时间被分配为 t_r , 则在 $t = t_r$ 时有 P_x 个事件已经被释放且正在驱动状态转移, 有 Q_x 个事件尚未驱动状态转移, 包括即将被释放的事件 a_i , 则 $N_x = Q_x + P_x$ 。

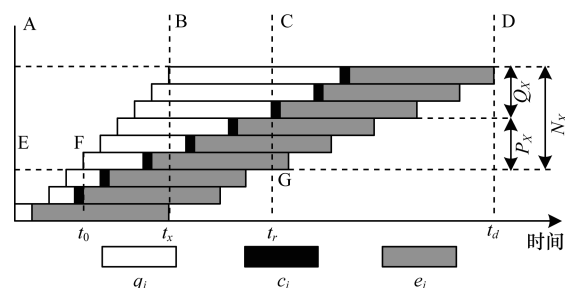


图 4 前馈预测模型

定义 7 令 T_x 表示驱动器输出事件 a_1 到事件 a_n 驱动状态 s_{n-1} 完成状态转移所用的时间。每一事件 a_i 从驱动器输出到完成驱动状态转移所用的平均时间为 $\bar{T}$, 则: $T_x = \bar{T} \times N_x$ 。

在图 4 中, $\bar{T}$ 表示每个小矩形的长, 即 $\bar{T} = q_i + c_i + e_i$ 。 $\bar{T}$ 与 N_x 所围成的面积为:

$$T_x \approx BFGD = (AEGC - AEFB) + CGD$$

观察图 4 可以看出:

$$\begin{cases} S_{AEGC} = t_r \times N_x \\ S_{AEFB} = \frac{1}{2} N_x (t_0 + t_x) \\ S_{CGD} = \frac{1}{2} N_x (t_d - t_r) \end{cases}$$

由定义 7 可得:

$$\begin{aligned} T_x &= \bar{T} \times N_x \approx (AEGC - AEFB) + CGD \\ &= t_r \times N_x - \frac{1}{2} N_x (t_0 + t_x) + \frac{1}{2} N_x (t_d - t_r) \end{aligned}$$

因此, 有:

$$\bar{T} \approx t_r - \frac{1}{2} (t_0 + t_x) + \frac{1}{2} (t_d - t_r) \quad (4)$$

定义 8 设事件从分配到释放时间到完成驱动状态转移的平均用时为 $\bar{\omega}$, 则 $(t_d - t_r) = \bar{\omega} \cdot \sum_{i=1}^{N_x} a_i$ 。

由定义 7 及定义 8 可得 $\bar{T} \approx t_r - \frac{1}{2} (t_0 + t_x) + \frac{1}{2} \bar{\omega} \cdot \sum_{i=1}^{N_x} a_i$, 则有 $\sum_{i=1}^{N_x} a_i \approx \frac{2}{\bar{\omega}} \{ \bar{T} - t_r + \frac{1}{2} (t_0 + t_x) \}$ 。

式(4)描述的即在事件队列中等待分配释放时间的事件累积量。所以, 未处理的事件量 Q_x 即为可重新分配释放时间的事件量, 分析用于前馈调度的前馈预测有效值参考量 $V_f(t)$ 。由定义 8 可知:

$$V_f(t) \approx \sum_{i=1}^{N_x} a_i - \sum_{i=1}^{P_x} a_i \approx \frac{2}{\bar{\omega}} \{ \bar{T} - t_r + \frac{1}{2} (t_0 + t_x) \} - \sum_{i=1}^{P_x} a_i$$

2.3 反馈控制器设计

反馈控制器(vt-FC)模型如图 5 所示。

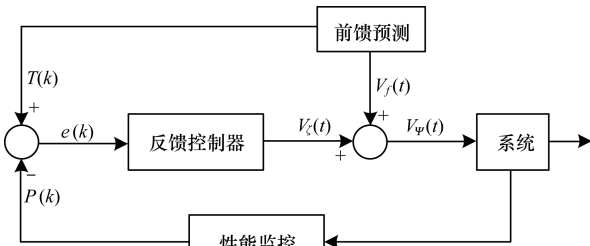


图 5 反馈控制器模型

vt-FC 通过对系统性能的连续监测, 得到状态转移性能 $P(k)$, 设 $P(k) = -|P(k)|$; 由事件队列通过 FFM 模型的目标状态转移性能 $T(k)$, 设 $T(k) = |T(k)|$, 经反馈监测模型(Feedback Monitoring Model, FMM)计算出误差:

$$e(k) = |T(k)| - |P(k)| \quad (5)$$

在 $n(n \geq 1)$ 个事件驱动状态转移后系统由性能监控器计算得出状态转移性能 $P(k)$, 该性能取决于此段状态转移周期时间内的状态转移性能 $P_s(k-i)$

及系统的平均状态转移性能 $P_b(k-i)$ 。设变量 α_i, β_i 满足:

$$P(k) = \sum_{i=1}^n \{ \alpha_i P_s(k-i) + \beta_i P_b(k-i) \}$$

由式(5)得到的误差可以计算出反馈控制器对系统状态转移性能控制的精确度:

$$R^2 = \frac{1 - \Delta(e(k))}{\Delta(t)} \quad (6)$$

其中, $\Delta(e(k))$ 为反馈控制器得到的误差 $e(k)$ 的变化量, $\Delta(t)$ 为系统运行反馈控制器的时间变化量。根据反馈控制器的精确度指标, 系统的 $R^2 \geq 0.8$ 则说明本控制器满足精确度要求。

定义 9 设反馈控制器的开环传递函数为 $G(z)$ $= \frac{T(z)}{V_\xi(z)}$, 得到闭环传递函数: $\Phi(z) = \frac{P(z)G(z)}{1 - P(z)G(z)}$, 令: $V_\xi(t) = V_\xi(t-1) + K_p[(K_I+1)e(k) - e(k-1)]$ 表示在反馈控制器中加入比例环节及积分环节, 即 PI 控制器。由于微分环节会增大系统噪音, 因此不考虑微分环节, 则最终的闭环传递函数为:

$$\Phi(z) = \frac{K_p(K_I+1)[z-1/(K_I+1)]}{z-1} \quad (7)$$

2.4 控制器实时操作

假设在超致密模型 $t = (\tau, n)$ 中, 随着释放时间的推移, 存在 $[(\tau_i - \tau_{i-1}) | \lim_{i \rightarrow \infty} = \infty] = [\lim_{i \rightarrow \infty} x_i = \infty] = [\lim_{i \rightarrow \infty} \nu_i = \lim_{i \rightarrow \infty} \nu_i^{\min}]$, 此时 $\lim_{i \rightarrow \infty} J_i(t) = 0$, 当此事件作用于相应状态时无意义。

通过 2.2 节与 2.3 节描述的控制器模型, 得到调度事件释放时间的有效值阈值 $V_\psi(t)$, 且计算出在 $t = ((n-1)\tau, n\tau)$ 时事件作用于状态的价值量是否满足 $(j_i^* \leq c | j_i^* \geq c)$ 。设在释放时间 t_r 释放的事件在到达相应状态且能满足价值量阈值的时刻为 d_i 。

1) 若 $d_i < t$, 说明在事件作用于状态前该事件已经失效, 则通过有效值阈值 $V_\psi(t)$ 调度事件队列, 将释放时间 t_r 向前调整为新的释放时间 t'_r , 以满足价值量阈值。

2) 若 $d_i > t$, 说明在事件作用于状态时该时间虽然满足价值量阈值, 但是对系统来说不满足性能最佳, 因此, 通过反馈控制器通知调度事件队列, 将释放时间 t_r 向后调整为新的释放时间 t''_r , 以满足系统性能最佳。

2.5 性能分析

本文通过调用 Matlab 的 RTC 工具包^[18]建立一种并行分布控制网络的实时信号输入输出模型^[6], 模拟实时控制系统的状态转移模型并通过观察模型对事件价值量的分析预测, 研究系统延时和系统响应时间, 验证 FF-PSM 模型是否能提高控制系统的实时性与确定性。实验在处理器为 Intel(R) Core(TM) i5-4590, CPU 为 3.30 GHz, RAM 为 8.00 GB 的 PC 机上运行, 使用 Matlab R2015a 进行仿真测试。

测试任务集在系统随机产生的 3 行 1 列的矩阵中随机抽取,随机变量在 1 个~200 个时间单元之间产生,并以 6% 的输出比率生成一个有 200 个可调度事件集的任务。

为得到一个响应速度最快的增益值,由式(7)构建数学模型,通过改变 K_p ,观察系统阶跃响应的变化,如图 6 所示,得到 $0 < K_p < 1.974 4$,当 K_p 在这一范围时,响应速度变快,而 $K_p > 1.974 4$ 后,振荡加剧,系统性能降低,因此,取 $K_p = 1.974 4$ 。

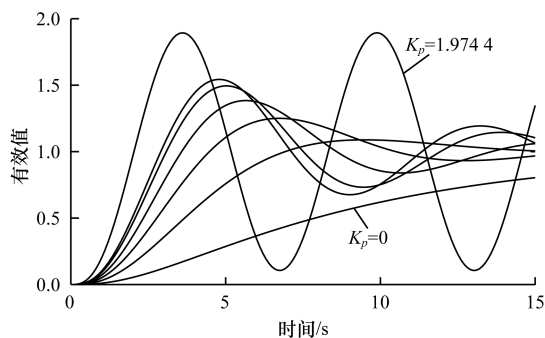


图 6 不同 K_p 下的系统阶跃响应曲线

同时建立 FFM 模型、FMM 模型与 FF-PSM 模型,在驱动器释放队列随机输入 200 个事件,事件通过 3 种模型调度后对系统性能的有效值对比如图 7 所示。可以看出,在事件输入量至 80 个后,系统性能波动趋于平稳,FFM 模型有前馈预测作用,故在事件量较小时,对系统的影响比 FMM 模型好,而在 FMM 模型反馈监测作用下,在事件量增大后,对系统的影响好于 FFM 模型,而 FF-PSM 模型对系统性能的影响在整个时间段内普遍优于 FFM 模型与 FMM 模型。

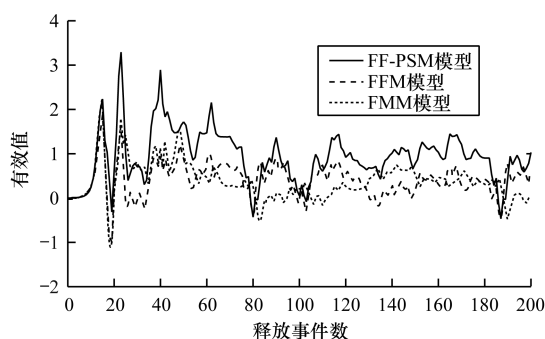


图 7 各模型对系统性能的有效值对比

选择 FF-PSM 模型随机重复 3 次实验分析释放事件量对系统延时的影响,预测系统延时为 0.9 s,3 次实验结果如图 8 所示。可以看出,在输入 80 个事件前,系统延时随事件量的增加迅速增大,在事件量达到 80 个之后,系统趋于稳定,系统延时不再增大,基本与预测延时保持一致,说明通过 FF-PSM 模型调整,能满足控制系统的实时性与确定性。

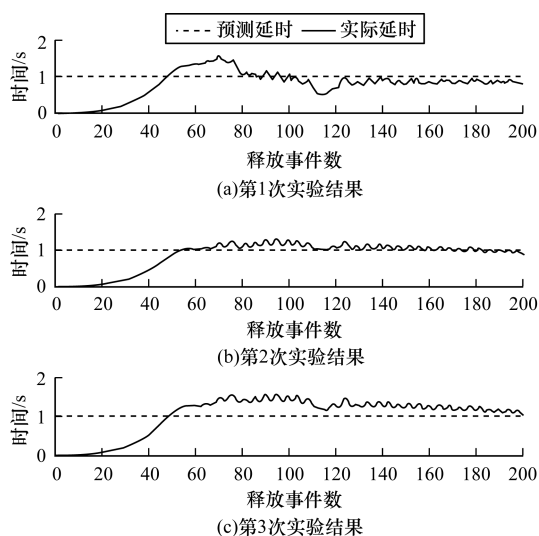


图 8 释放事件量对系统延时的影响

3 结束语

本文通过建立前馈预测与反馈调节相结合的预测调度模型 FF-PSM,描述了含价值量约束与时钟约束的超致密时间模型下嵌入式实时控制系统状态转移的过程,使控制系统性能满足最优实时性与确定性。首先定义事件由队列释放至相应状态并驱动其转移的过程,研究事件驱动状态转移的约束条件与事件驱动状态转移在超致密时间模型下的确定性,然后利用价值量有效值的特性构建前馈预测与反馈控制相结合的调度事件释放顺序模型,最后由 FF-PSM 模型介绍模型控制器实时操作。通过比较系统有效值和延迟时间,表明该模型能在一定程度上提高控制系统的实时性与确定性。

参考文献

- [1] 王剑平,张云生,张 果,等. 实时系统时间并行模型确定性分析[J]. 控制与决策,2012,27(12):1859-1863.
- [2] 王剑平,张云生,张 果,等. 嵌入式控制状态转移的确定性实时语义[J]. 控制与决策,2014,29(1):83-88.
- [3] 邹荣念,赵文涛,王鹏飞,等. 嵌入式系统可生存性分析模型研究[J]. 计算机工程,2013,39(9):178-182.
- [4] 刘 静. 应用混合队列调度策略的 AFDX 实时性优化研究[J]. 计算机工程,2015,41(11):135-141.
- [5] 张 晶,陈沫良. 嵌入式软件算法级能耗建模与分析[J]. 计算机工程,2014,40(6):13-15.
- [6] Bengtsson J, Wang Yi. Timed Automata: Semantics, Algorithms and Tools [M]//Desel J, Reisig W, Rozenberg G. Lectures on Concurrency and Petri Nets. Berlin, Germany: Springer,2004:87-12.
- [7] 王剑平,张云生,张 果,等. 并行分布控制网络的实时信号时序流图分析[J]. 控制与决策,2010,25(11):1727-1731.

(下转第 100 页)

表 1 部分动态库轻量化结果对比 MB

动态库	原有大小	轻量化大小
libcontent. so	229.00	14.20
libevents. so	1.21	0.12
libipc. so	1.99	0.18
libbase. so	14.50	1.30

6 结束语

随着互联网技术的高速发展,嵌入式系统智能化程度越来越高,智能终端对于应用的安全性提出了更高的要求。随着智能终端对复杂计算处理的要求越来越高,需要对应用进行更加高效的管理。为实现浏览器对 Web 应用的高效管理和安全隔离,本文提出一种基于 Chromium 的渲染进程轻量化隔离方法,分析了 Chromium 的多进程架构以及主进程对渲染进程的管理机制,包括进程的创建、进程的优先级处理、进程的关闭以及进程间通信等。介绍了 Docker 虚拟化容器技术及优势,给出与 Docker 相结合的渲染进程轻量化隔离方案,以及 ZygoteDocker 的技术方案设计、功能设计与渲染进程分离的设计,并基于此进行系统的初步实现和测试,结果表明,该方法渲染进程模块在容器内实现了隔离,轻量化结果较为明显。

参考文献

- [1] Malinowski A, Yu H. Comparison of Embedded System Design for Industrial Applications [J]. IEEE Transactions on Industrial Informatics, 2011, 7(2): 244-254.
- [2] Xie X W, Mao H, Xie X W, et al. Mobile Intelligent Terminal Equipment Learning Software Design and Development [C]//Proceedings of International Conference on Intelligent Transportation, Big Data & Smart City. [S. l.]: IEEE Computer Society, 2015: 515-518.
- [3] 赵波, 夏忠林, 安杨, 等. 用于虚拟化环境的进程隔离方法研究与实现[J]. 华中科技大学学报(自然科学版), 2014(11): 74-79.

- [4] 李瑜, 赵勇, 梁鹏. 面向服务进程的用户权限隔离模型[J]. 计算机工程, 2011, 37(23): 141-143.
- [5] Huang D, He B, Miao C. A Survey of Resource Management in Multi-tier Web Applications [J]. IEEE Communications Surveys & Tutorials, 2014, 16(3): 1574-1590.
- [6] 刘秀秀, 潘梁, 郭志川, 等. 基于 Web 运行环境的 Android 原生应用管理研究[J]. 网络新媒体技术, 2015, 4(4): 35-40.
- [7] Kristinesønn S R. A Study of Linux Containers and Their Ability to Quickly Offer Scalability for Web Services [D]. Oslo, Kongeriket Norge: University of Oslo, 2015.
- [8] Chen W, Xu L, Li G, et al. A Lightweight Virtualization Solution for Android Devices [J]. IEEE Transactions on Computers, 2015(1): 2741-2751.
- [9] Bernstein D. Containers and Cloud: From LXC to Docker to Kubernetes [J]. IEEE Cloud Computing, 2014, 1(3): 81-84.
- [10] Reis C. Process Models [EB/OL]. (2013-01-15). <https://www.chromium.org/developers/design-documents/process-models>.
- [11] Wilson B, Goodger B. Multi-process Architecture [EB/OL]. (2015-04-30). <https://www.chromium.org/developers/design-documents/multi-process-architecture>.
- [12] Harris W. Inter-process Communication [EB/OL]. (2015-11-13). <https://www.chromium.org/developers/design-documents/inter-process-communication>.
- [13] Nider J, Rapoport M. Cross-ISA Container Migration [C]//Proceedings of ACM International on Systems and Storage Conference. New York, USA: ACM Press, 2016: 259-264.
- [14] Hacker T J, Romero F, Nielsen J J. Secure Live Migration of Parallel Applications Using Container-based Virtual Machines [J]. International Journal of Space-based and Situated Computing, 2012, 2(1): 45-57.
- [15] Vinoski S. Where is Iddleware [J]. IEEE Internet Computing, 2002, 6(2): 83-85.
- [16] Maurer W D, Lewis T G. Hash Table Methods [J]. ACM Computing Surveys, 1975, 7(1): 5-19.

编辑 索书志

(上接第 94 页)

- [8] Ramchandani C. Analysis of Asynchronous Concurrent Systems by Timed Petri Nets [D]. Cambridge, USA: Massachusetts Institute of Technology, 1973.
- [9] 刘昊知, 胥布工, 高福荣. 基于时序模型迁移的间歇过程监测建模方法[J]. 控制与决策, 2015, 30(10): 1885-1889.
- [10] 黄妹娟, 朱怡安, 李兵哲, 等. 具有依赖关系的周期任务实时调度方法[J]. 计算机学报, 2015, 38(5): 999-1006.
- [11] Oh J, Kyoung-Don K. A Predictive-reactive Method for Improving the Robustness of Real-time Data Services [J]. IEEE Transactions on Knowledge and Data Engineering, 2013, 25(5): 974-986.
- [12] Zhou Yan, Kang K D. Deadline Assignment and Feedback Control for Differentiated Real-time Data Services [J]. IEEE Transactions on Knowledge and Data Engineering, 2015, 27(12): 3245-3257.

- [13] Alur R, Dill D L. A Theory of Timed Automata [J]. Theoretical Computer Science, 1994, 126(2): 183-235.
- [14] 赵昱, 何锋, 徐亚军, 等. 时间触发总线流量调度机制及其实时性分析[J]. 计算机工程, 2015, 41(10): 59-65.
- [15] 张彬连, 徐洪智. 基于随机任务的可靠性约束与节能调度算法[J]. 计算机工程, 2015, 41(8): 1-5.
- [16] Song Han, Lam K, Wang Jiantao. Adaptive Co-scheduling for Periodic Application and Update Transactions in Real-time Database Systems [J]. Journal of Systems and Software, 2012, 85(8): 1729-1743.
- [17] 黄国兵, 李瑞玲, 李华丽, 等. μ C/OS-II 任务优先级调度算法分析与改进[J]. 计算机工程, 2015, 41(8): 52-54, 60.
- [18] Ernesto W, Lothar T. Real-Time Calculus (RTC) Toolbox [EB/OL]. [2016-03-20]. <http://www.map.ethz.ch/Rtctoolbox>.

编辑 金胡考