

## 基于总线的开放式控制系统调度策略

汤峰, 张平, 李方, 黄致祥

(华南理工大学 计算机科学与工程学院, 广州 510640)

**摘要:** 提出“基于改进EDF的模糊反馈调度策略”, 并应用于基于总线的开放式控制系统. 设计了一个适用于此类总线控制架构的任务调度方案, 由任务截止期、任务重要性以及传输误差和响应时间的综合影响来动态调节任务优先级, 用以解决由引入总线而导致的响应速度慢、总线阻塞时的丢包问题. 通过对机器人的控制实验验证了所提出调度策略的可行性. 实验结果表明, 加入调度模块后控制系统的响应时间和丢包率都有显著改善, 系统的实时性得到了提高.

**关键词:** 总线; 机器人控制系统; 调度策略; 最早时限优先; 模糊反馈调度

**中图分类号:** TP273

**文献标志码:** A

## Strategy of scheduling in open control systems over bus

TANG Feng, ZHANG Ping, LI Fang, HUANG Zhi-xiang

(School of Computer Science & Engineering, South China University of Technology, Guangzhou 510640, China.

Correspondent: ZHANG Ping, E-mail: zhangp@scut.edu.cn)

**Abstract:** In order to solve the problem of slow response and packet loss which are caused by the introduction of the bus in the robot control system, a task scheduling scheme is designed, and the scheduling policy based on improved fuzzy feedback EDF is proposed, which adjusts dynamically the task priority according to the combined effects of the response time and transmission errors. The feasibility of scheduling strategy is validated through the robot control experiments. The results show that the performance of system response time and packet loss ratio can be improved significantly after adding scheduling module, and the real time performance of the system is also improved.

**Keywords:** the bus; robot control system; scheduling scheme; earliest deadline first(EDF); fuzzy feedback scheduling

## 0 引言

在实时性要求较高的控制系统中, 实时调度理论是非常关键的研究内容. 基于总线的开放式控制系统<sup>[1]</sup>将各功能模块以服务构件的形式连接到统一的总线上, 实现高度的松耦合结构<sup>[2]</sup>. 但是由于总线的介入, 控制流程中的各模块不能直接交互而影响控制实时性, 引起控制时延、数据丢包及抖动等现象. 时延、丢包等时间不确定性问题可能引起控制性能下降或系统不稳定, 严重时甚至导致工业生产的故障<sup>[3-4]</sup>. 文献[5]提出了控制系统的控制误差不仅取决于控制算法的精确度, 还依赖于控制系统的性能指标. 控制与任务调度虽属于不同领域, 但控制的时间参数与调度的时间属性是相关联的, 时延和丢包是关联控制性能和任务调度实施的关键时间参数, 时延和丢包数越小, 控制系统服务品质越好. 因此, 有必要采用调度与

控制协同设计的方法<sup>[6]</sup>, 在控制系统设计的同时考虑总线消息调度优化问题, 以减小时延, 提高系统响应的实时性, 实现系统的总体性能优化.

控制与调度协同设计的方法是Seto等<sup>[7]</sup>最早提出的, 该方法考虑了控制性能(QoC)和服务品质(QoS)的综合性能优化, 已成为协同设计的研究热点<sup>[8-9]</sup>. 在Liu等<sup>[10]</sup>提出的单调速率(RM)和最早时限优先(EDF)实时调度算法的基础上, 国内外学术界进行了扩展研究. Branky等<sup>[11]</sup>把RM算法应用于网络控制系统调度, 对NCS的传感器节点数据传输进行调度, 研究了稳定性约束下的最优调度方法; Brockett<sup>[12]</sup>针对资源受限NCS的多个控制环调度问题提出了通信序列的思想, 离线确定网络上各控制环使用网络的次序, 并将之作为一个通信序列, 在调度中按照该序列重复执行; 最大误差优先LEF<sup>[13]</sup>和MEF-

收稿日期: 2014-03-05; 修回日期: 2014-07-31.

基金项目: 广州市科技计划项目(2013Y2-00100); 中央高校基本科研业务费资金项目(2014ZM0048).

作者简介: 汤峰(1979-), 女, 博士生, 从事机器人控制的研究; 张平(1964-), 男, 教授, 博士生导师, 从事机器人控制等研究.

TOD<sup>[14-15]</sup>调度是具有代表性的基于误差优先级的反馈调度方法,具有最大误差的节点将优先使用网络资源;文献[16]根据不同控制回路重要性的不同,对各控制回路的误差赋予不同的权重,提出了一种基于反馈的最大紧急度优先的反馈调度算法;文献[17]提出了一种基于误差的采样周期调度算法,根据误差的大小动态分配网络带宽;文献[18]提出了基于系统稳定性的调度算法;文献[19]针对网络控制系统采用了基于数学规划方法的优化调度算法,控制性能的优化函数和性能函数均表示为采样周期是 $h_i$ 的函数,采样周期值的获取转换为了多目标优化求解问题<sup>[20]</sup>;Gaid等<sup>[21-22]</sup>对资源受限的多变量NCS采用MLD进行建模,并转换为混合整型二次规划问题,对任务周期进行优化求解。

本文采用控制与调度协同设计的研究方法,在设计控制系统的同时,考虑总线消息调度优化问题,把时延、丢包作为控制调度考虑的时间性能参数来构建调度模型,提高系统响应的实时性,实现系统的总体性能优化。传统“开环”的实时调度实现较为简单,响应速度较快,但是不能根据系统的变化及时调整调度目标。引入闭环反馈策略是对开环实时调度中的一个重大改进,是实时调度研究的新方向。

## 1 调度模块框架

借鉴实时网络调度的研究经验,结合文献[1]中开放式控制系统设计架构,综合考虑传输误差及响应时间的影响,提出“基于改进EDF的模糊反馈调度策略”。该策略的基本思想是:任务消息 $T$ 到达总线时先根据改进的EDF调度算法分配优先级,当调度采样周期到来时再根据反馈信息(传输误差与响应时间的加权)对各任务优先级进行调整,达到对系统性能恶化的抑制。该调度策略同样适用于基于CAN总线、CORBA等开放式网络控制系统。

控制系统任务调度主要考虑如何设置执行各控制任务的先后次序以满足系统的实时性。系统中时变的负载导致了不可预期的运行环境,需要有效处理不确定的资源对控制性能的影响,因此采用反馈式调度策略。反馈调度的特点是:以系统丢包率和响应时间比的多目标函数作为测量标准,通过“基于改进EDF的模糊反馈调度策略”分配各任务的优先级,达到对系统性能恶化的抑制。

图1所示为反馈调度模块结构,在一个调度采样周期内,第 $i$ 个任务控制回路的反馈性能指标函数为

$$J_i(j) = \alpha E_i(j) + \beta \tau_i(j).$$

其中

$$E_i(j) = \frac{X_L(j)}{X_L(j) + X_S(j)},$$

$$\tau_i = \frac{R(j)}{R_{\max}}.$$

$X_L(j)$ 表示该任务回路中传送不成功的数据包总数, $X_S(j)$ 表示该任务回路中成功传送数据包的总数; $R(j)$ 表示一个任务流程的响应时间, $R_{\max}$ 表示任务最大的响应时间; $\alpha$ 和 $\beta$ 分别表示丢包误差率和响应时间比的权值,有 $\alpha + \beta = 1$ ;  $JC_i(j) = J_i(j) - J_i(j-1)$ 为第 $i$ 个任务回路性能指标的差分,反应性能指标的变化方向。性能指标函数通过对丢包误差率和响应时间比的加权求和转化为单目标优化问题。权值需要根据实际系统要求或通过仿真方法确定,如果响应时间充裕,则使丢包误差的权重值增大;如果响应时间紧迫,则使响应时间的权重值增大。

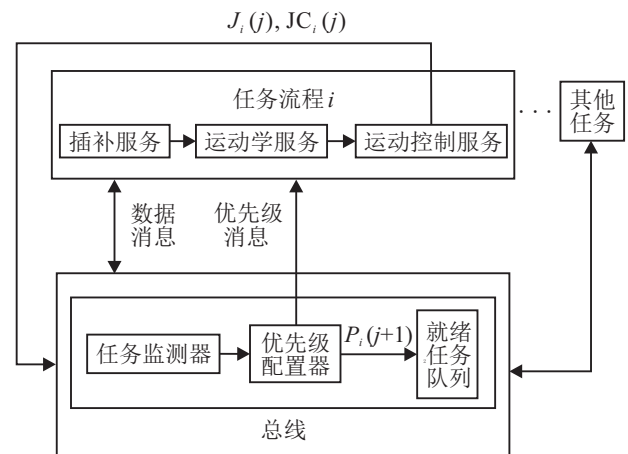


图1 反馈调度模块结构

若系统中有 $n$ 个任务在运行,调度采样周期 $h_f$ 应等于 $n$ 个任务中最长任务周期的整数倍,有

$$h_f = A \max[h_i].$$

其中: $A$ 为整数, $h_i$ 为第 $i$ 个任务的周期, $i \in [0, n]$ 。为了避免频繁的反馈调度切换影响系统的整体性能, $A$ 一般大于1。假设 $h_i$ 为最长的任务周期。任务流程包括:插补子任务、运动学子任务和运动控制子任务。各任务周期最大值为 $h_i$ ,考虑到流程的完整性, $A$ 取值为3,则 $h_f = 3h_i$ 。

调度器采用时间驱动方式,当采样时刻到来时,调度器监测各任务运行状态,获取以下信息:

- 1) 各任务在总线上传输失败的数据包数量 $X_L(j)$ 及任务结束时成功传输的数据包数量 $X_S(j)$ ;
- 2) 各任务的指令响应时间 $\tau$ 及任务的数据包消息在总线中的转发时间 $\tau_b$ 。

在调度与控制协同设计中,系统任务优先级往往依据各任务回路反馈的性能来进行动态分配和调节,任务控制回路的性能越差,该任务的优先级也就越高。下面对系统应用层上的任务调度进行描述。

## 2 任务调度策略

### 2.1 系统优先级规划与映射

引入了双层调度器的设计思想, 这种分层设计的好处是实现了总线调度模块的平台无关性. 第1层调度只负责全局优先级的分配, 与具体操作系统的调度机制无关; 第2层调度器与操作系统的底层调度算法密切相关, 负责将全局优先级映射到系统的有效优先级. 当需要将开放式控制系统移植到其他操作系统平台时, 只需要对第2层调度进行修改. 可见, 这种双层调度框架使得系统的可移植性得到很大增强.

采用 Windows+RTX 的通用操作系统实时扩展模式, 并不直接修改 RTX 平台的调度模块, 而是通过优先级映射的方式, 给作业线程分配优先级. 在 RTX 内核子系统中, 线程可以划分为 0~127 个优先级, 共有 128 个优先级. 整个控制系统优先级原则为:  $P$  紧急事件 >  $P$  调度模块 >  $P$  总线其他线程 >  $P$  其节点上其他各服务. 系统需要监控运行的状态, 当节点中有新的服务或线程启动时, 必须重新对系统进行优先级映射.

### 2.2 任务优先级分配

本文采用以下调度算法为各任务分配优先级, 并根据优先级把任务消息 (消息中包含此任务到达时间、截止期等优先级属性) 排列到分发就绪队列中, 以任务消息的分发促发各子任务的执行. 在一个任务周期  $h_i$  中, 各子任务优先级属性应保持一致.

由于传统的 EDF 算法不能对未知运行环境下的任务作出及时调整, 而由于引入反馈机制, 反馈调度会导致调度的不及时 (调度周期不宜太短, 避免任务优先级的频繁切换). 本文提出“基于改进 EDF 的模糊反馈调度策略”分配各任务的优先级, 并按照全局优先级递增的顺序在就绪任务队列中进行排序. 基本思路如图 2 所示.

1) 当一个任务消息  $T$  到达总线时, 为它分配初始优先级属性值, 并按照式 (1) 计算优先级, 插入到就绪任务队列中. 任务记为  $T(\alpha, de, \varepsilon, i)$ , 其中属性  $\alpha$ 、 $de$ 、 $\varepsilon$  和  $i$  分别表示任务的到达时间、截止期、反馈因子及任务重要性. 相对截止期可定义成任务的截止期与当前时刻之差 (即  $d = de - t$ ), 也是随时间变化的. 对于具有不同的典型相对截止期  $d_i$ 、典型反馈因子  $\varepsilon_j$ 、典型任务重要性  $i_k$  的任务  $T_{ijk}$ , 赋予其优先级值为  $P(T_{ijk}, d_i, \varepsilon_j, i_k)$ , 其中:  $i = 1, 2, \dots, m$ ,  $j = 1, 2, \dots, n$ ,  $k = 1, 2, \dots, r$ , 简记为  $P_{ijk}$ .

引入反馈因子  $\varepsilon_j$  作为确定任务优先级的衡量标准之一, 与任务重要性  $i$  和相对截止期  $d$  共同决定任

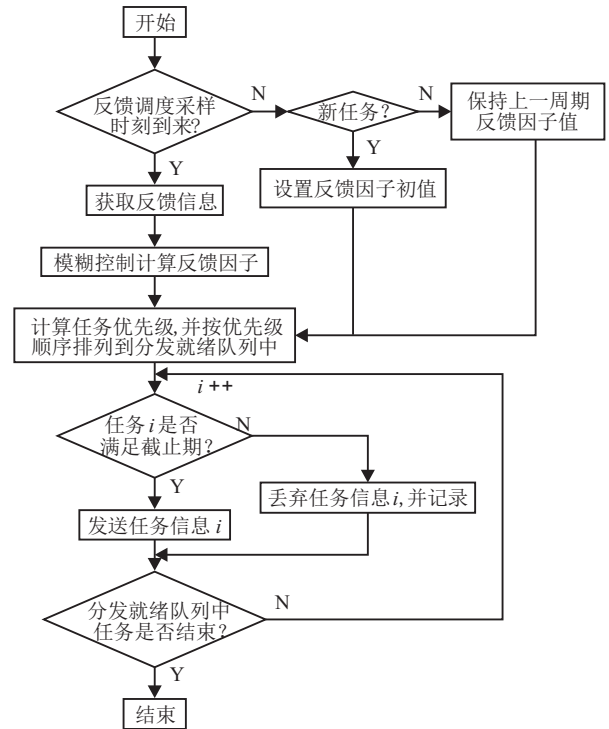


图 2 调度流程

务  $T$  的优先级, 一般表示为

$$P(T, d, J) = \zeta \varepsilon \times i/d. \quad (1)$$

设任务的初始反馈因子  $\varepsilon = 0.5$ , 调节系数  $\zeta$  需要通过仿真方法确定. 这样算出的优先级可能存在多个任务同一优先级的情况, 同一优先级任务可以再根据反馈性能指标或截止期排序. 由式 (1) 可知, 截止期越早, 反馈因子越大, 任务重要性越大, 优先级越高, 反之亦然. 这样做的优点是全面考虑了任务本身的多方面属性, 并且由多个参数体现在优先级中, 弥补了传统 EDF 算法仅考虑截止期带来的不足.

任务中各子任务均拥有相同的截止期优先级属性, 这样做的好处是: 由于系统同一任务流程的各子任务之间是异步的, 当任务  $T(j)$  的最后一个子任务消息和  $T(j+1)$  的子任务消息同时到达总线时, 因为  $T(j)$  的截止期先于  $T(j+1)$ , 保证了  $T(j)$  最后一个子任务消息优先级高于  $T(j+1)$  的子任务消息, 也就是越接近硬件的子任务优先级越高的原则, 保证了系统的稳定性.

2) 当调度采样周期到来时, 根据反馈信息 ( $J_i(j)$ 、 $JC_i(j)$ ), 采用模糊调度规则计算各任务反馈因子, 并把反馈因子值代入式 (1) 中计算任务优先级. 模糊计算具体算法如下.

采用模糊控制领域较为成熟的 Mamdani 模糊处理方法, 其处理过程大致可分为模糊化、模糊推理、模糊合成和解模糊化 4 个步骤.

模糊反馈调度模块输入为反馈性能指标函数  $J_i(k)$  和性能指标的差分  $JC_i(k) = J_i(k) - J_i(k-1)$ , 经模糊化为  $J, JC$ , 它们的模糊集为  $\{NB, NS, ZE, PS, PB\}$ . 输出变量  $\varepsilon$  的模糊集为  $\{S, M, B\}$ , 模糊控制器的控制规则如表 1 所示.

表 1 模糊控制规则

$\varepsilon$	JC				
	NB	NS	ZE	PS	PB
NB	B	B	B	B	B
NS	B	M	S	M	B
ZE	M	S	S	S	M
PS	B	M	S	M	B
PB	B	B	B	B	B

建立模糊规则的基本思想是: 当控制任务的误差性能及差分同向且其绝对值较大时, 说明控制系统正远离性能指标设定值并有继续偏离的趋势, 这意味着系统性能较差, 此时反馈因子取值较大; 当控制任务的误差性能及差分较小时, 说明系统性能在设定值附近, 性能较好, 此时反馈因子取值较小; 当控制任务的误差性能及差分反向时, 说明系统运行轨迹正向设定值靠近, 这意味着系统性能有好的发展趋势, 此时反馈因子可以取得相对小一些.

去模糊化可采用重心法得到反馈因子, 有

$$\varepsilon = \frac{\sum_{i=1}^n \varepsilon_i \mu(\varepsilon_i)}{\sum_{i=1}^n \mu(\varepsilon_i)}. \quad (2)$$

得到反馈因子后, 总线需要把该值传送给对应的任务, 以供该任务的下一周期消息调度使用.

```
Now adjusting the list!!
```

```
size = 6
```

```
1: service ID = 16202      deadline = 129964974490228663
2: service ID = 30186     deadline = 129964974493067141
3: service ID = 7363      deadline = 129964974488790034
4: service ID = 8679      deadline = 129964974491642807
5: service ID = 12856     deadline = 129964974491628663
6: service ID = 9832      deadline = 129964974494505122
```

```
Scheduling !!!!!!!
```

```
service 7363 : priority = 0.018 733
service 16202 : priority = 0.017 710
service 12856 : priority = 0.016 687
service 8679 : priority = 0.017 632
service 30186 : priority = 0.016 609
service 9832 : priority = 0.015 589
```

```
size = 6
```

```
1: service ID = 7363      deadline = 129964974491628663
2: service ID = 16202     deadline = 129964974493067141
3: service ID = 12856     deadline = 129964974488790034
4: service ID = 8679      deadline = 129964974494505122
5: service ID = 30186     deadline = 129964974491628663
6: service ID = 9832      deadline = 129964974491642807
```

图 4 调度实例

### 3 控制系统调度模块的实验研究

将调度模块应用于 6 自由度机械臂的控制原型系统, 实验操作对象为固高科技六自由度机器人 (GRB3016 型), 如图 3 所示.



图 3 实验操作对象

#### 3.1 调度实例

实验中开启两个控制任务流程 TASK 1 和 TASK 2, 分别对两个机械臂进行控制, TASK 1 重要性高于 TASK 2. 每个控制任务流程中包含插补任务、运动学变换任务 (正、反解运算) 和运动控制任务.

调度实例如图 4 所示, 此时总线队列中有 6 个待转发的数据包, TASK 1 控制任务流程包含: service ID 为 8679 的插补任务消息、service ID 为 16202 的反解任务消息和 service ID 为 7363 的运动控制任务消息. TASK 2 控制任务流程包含: service ID 为 9832 的插补任务消息、service ID 为 30186 的反解任务消息、service ID 为 12856 的运动控制任务消息. 最上方为调度前的排列情况, 最下方为调度后的排列情况.

从图4可以看出: 由于TASK 1控制任务流程的重要性比较高, 即使它的截止期大, ID为16202和7363仍然排在TASK 2所有消息前面; TASK 2的ID为12856的运动控制任务因为临近截止期, 所以优先级高于TASK 1 ID为8679的任务; 而在同一个流程中, 运动控制任务的消息数据包截止期比较小, 所以优先级比较高, 截止期靠后的数据包会排在后边, 这就保证了越接近硬件的消息能够越优先得到转发。

### 3.2 实时性测试

在总线上有调度模块和无调度模块两种情况下进行10组数据的测试, 记录了控制流程的响应时间和总线的处理时间, 如图5和图6所示。其中: 带■的直线表示无调度的平均响应时间, 带▲的直线表示有调度的平均响应时间。

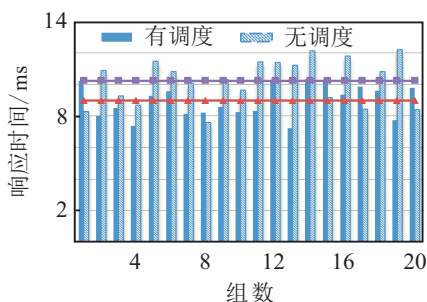


图5 控制流程响应时间

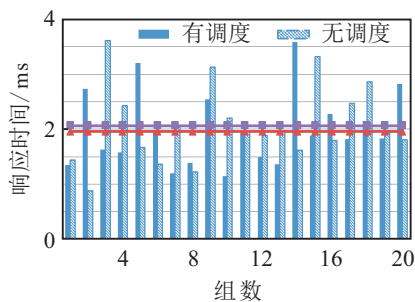


图6 控制流程总线处理时间

从图5和图6可以看出: 控制流程在有调度模块的总线下的平均响应时间为8.973 ms, 在无调度模块的总线下的平均响应时间为10.2387 ms, 在有调度模块的总线下平均响应时间明显更短; 而且, 在各个服务与总线之间的传输时间上, 有调度模块的总线基本上都比无调度模块的总线更短, 说明了调度模块能在一定程度上缓解在高负载情况下各服务之间的传输阻塞延时, 同时配合本项目开发的接口协议及通信模式, 保证了系统的实时性。

### 3.3 丢包率测试

测试环境: 启动两个控制任务流程, 并设置任务流程截止期限为20 ms。总线丢包率测试数据如图7所示, 其中: 带■的直线表示无调度的平均丢包率, 带▲的直线表示有调度的平均丢包率, 平均丢包率为

各组丢包率之和减去一个最大值以及一个最小值的平均值。

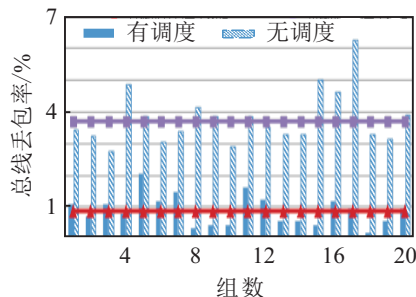


图7 总线丢包率测试

从图7可以看出, 无调度模块的总线丢包率为3.69%, 有调度模块的总线丢包率为0.85%, 有调度模块的总线丢包率明显较低, 从而表明调度模块能有效地降低系统的丢包率。

## 4 结论

双层反馈式调度框架适用于基于总线架构的开放式控制系统。总线上采用改进EDF的模糊反馈调度策略, 根据任务的截止期、重要性及反馈的性能指标动态调节任务的优先级, 使得重要性高的任务流程数据包发送的优先级高, 同一任务流程越接近硬件的数据包转发的优先级越高。控制回路性能指标恶化也会相应提高其优先级别, 达到对系统性能恶化的抑制。实验结果表明, 本文所提出的调度策略能够有效缓解机器人控制系统由于引入总线带来的响应速度慢、总线阻塞时丢包的问题。

### 参考文献(References)

- [1] Tang Feng, Zhang Ping. Open robot control platform based on LSOA[J]. Int Symposium on Robotics and Applications, 2013, 24(2): 734-744.
- [2] 苏彦彬. 基于SOA的开放式控制平台关键技术的研究[D]. 广州: 华南理工大学计算机科学与工程学院, 2011: 5-10.  
(Su Y B. Key technologies for implementing open control platform based on SOA[D]. Guangzhou: School of Computer Science & Engineering, South China University of Technology, 2011: 5-10.)
- [3] Kushner H, Tobias L. On the stability of randomly sampled systems[J]. IEEE Trans on Automatic Control, 1969, 14(4): 319-324.
- [4] Kalman R E, Bertram J E. A unified approach to the theory of sampling systems[J]. J of the Franklin Institute, 1959, 267(5): 405-436.
- [5] 张春华. 考虑任务实施时间因素的CNC系统建模与控制[D]. 广州: 华南理工大学机械与汽车工程学院, 2014: 43-53.  
(Zhang C H. Control and modeling of CNC system

- considering timing parameters for task implementaion[D]. Guangzhou: School of Mechanical & Automotive Engineering, South China University of Technology, 2014: 43-53.)
- [6] 赵维佳. 网络化运动控制系统资源调度研究[D]. 广州: 华南理工大学机械与汽车工程学院, 2010: 7-11. (Zhao W Q. Study on resource scheduling of networked motion control systems[D]. Guangzhou: School of Mechanical & Automotive Engineering, South China University of Technology, 2010: 7-11.)
- [7] Seto D, Lehoczkj P, Sha L, et al. On task schedulability in real-time control systems[C]. Proc of the 17th IEEE Real Time Systems Symposium. Washington: IEEE, 1996: 13-21.
- [8] Shi-Lu Dai, Hai Lin, Shuzhi Sam Ge. Scheduling-and-control codesign for a collection of networked control systems with uncertain delays[J]. IEEE Trans on Control Systems Technology, 2010, 18(1): 66-78.
- [9] Zhi-Wen Wang, Hong-Tao Sun. Control and scheduling co-design of networked control system: Overview and directions[C]. Proc of the 2012 Int Conf on Machine Learning and Cybernetics. Xi'an, 2012: 816-824.
- [10] Liu C L, Layland J W. Scheduling algorithms for multi-programming in a hard-real-time environment[J]. J of the Association for Computing Machinery, 1973, 20(1): 46-61.
- [11] Branicky M S, Phillips S M, Zhang Wei. Scheduling and feedback co-design for networked control systems[C]. Proc of IEEE Conf on Decision and Control. LasVegas: IEEE, 2002: 1211-1217.
- [12] Brockett R. Stabilization of motor networks[C]. Proc of the 34th IEEE Conf on Decision and Control. New Orleans: IEEE, 1995: 1484-1488.
- [13] Yopez J, Marti P, Fuertes J M. Control loop scheduling paradigm in distributed control systems[C]. Proc of the 29th Annual Conf of the IEEE Industrial Electronics Society. Roanoke: IEEE, 2003: 1441-1446.
- [14] Walsh G C, Ye Hong, Bushnell L G. Stability analysis of networked control systems[J]. IEEE Trans on Control Systems Technology, 2002, 10(3): 438-446.
- [15] Walsh G C, Beldiman O, Bushnell L G. Asymptotic behavior of nonlinear networked control systems[J]. IEEE Trans on Automatic Control, 2001, 46(7): 1093-1097.
- [16] Xia Feng, Dai Xiaohua, Wang Zhi, et al. Feedback based network scheduling of networked control systems[C]. Int Conf on Control and Automation. Budapest: IEEE, 2005: 1231-1236.
- [17] Velasco M, Fuertes J M, Lin C, et al. A control approach to bandwidth management in networked control systems[C]. The 30th Annual Conf of IEEE Industrial Electronics Society. Busan: IEEE, 2004: 2343-2348.
- [18] Branicky M S, Phillips S R, Zhang W. Stability of networked control systems: Explicit analysis of delay[C]. Proc of the American Control Conf. Chicago: IEEE, 2000: 2352-2357.
- [19] Velaseo M, Marti P, Frigola M. Bandwidth management for distributed control of highly articulated robots[C]. Proc of the 2005 IEEE Int Conf on Robotics and Automation. Bareelona: IEEE, 2005: 265-270.
- [20] Cervin Anton, Eker Johan, Bernhardsson Bo, et al. Feedback-feedforward scheduling of control tasks[J]. Real Time Systems, 2002, 23(1): 25-53.
- [21] El Mongi Ben Gaid M, Cela A, Hamam Y. Optimal integrated control and scheduling of systems with communication constraints[C]. Proc of the 44th IEEE Conf on Decision Control and European Control Conf. Seville: IEEE, 2005: 854-859.
- [22] Gaid M E M B, Cela A, Haman Y. Optimal integrated control and scheduling of networked control system with communication constrains: Application to a car suspension system[J]. IEEE Trans on Control System Technology, 2006, 14(4): 776-787.

(责任编辑: 齐 霖)