

文章编号: 1002-0411(2002)04-289-06

## FF 现场总线系统中实时任务的建模与调度

王天然<sup>1</sup> 周悦<sup>1,2</sup> 于海斌<sup>1</sup> 苑明哲<sup>1</sup>

(1. 中国科学院沈阳自动化研究所 沈阳 110016; 2. 沈阳建筑工程学院 沈阳 110015)

**摘要:** 本文研究了单网段 FF 现场总线系统中具有时间约束和次序约束的实时任务, 即功能块任务和通信任务的建模与调度. 首先, 将功能块任务和通信任务等视为相同的任务, 在只考虑任务间次序约束的情况下, 提出了基于紧凑模式的任务模型, 以保证每个作业被尽可能早地完成. 其次, 考虑单网段通信任务共享一个传输介质而引起的通信超时, 提出了基于作业速率单调优先级算法的扩展紧凑模式的任务调度算法, 以满足实时任务的时间约束和次序约束. 最后, 通过一个应用实例来描述实时任务的调度过程.\*

**关键词:** 实时; 任务; 通信超时; 紧凑模式; 作业速率单调

中图分类号: TP13

文献标识码: B

### MODELING AND SCHEDULING FOR REAL-TIME TASKS IN FF SYSTEM

WANG Tian-ran<sup>1</sup> ZHOU Yue<sup>1,2</sup> YU Hai-bin<sup>1</sup> YUAN Ming-zhe<sup>1</sup>

(1. Shenyang Institute of Automation Chinese Academy of Sciences, Shenyang 110016;

2. Shenyang Architecture and Civil Engineering Institute, Shenyang 110015)

**Abstract:** This paper presents modeling and scheduling for real-time tasks with time constraints and precedence constraints, i.e., Function Blocks tasks and communication tasks, in a single segment FF system. Firstly, Function Blocks tasks and communication tasks are regarded as same tasks, and only considering precedence constraints, we propose a compact mode to build model of tasks, in which each job is completed as earlier as possible. Secondly, considering communication overtime, a extended compact mode scheduling algorithm based on job rate monotonic (JRM) is presented to schedule all real-time tasks in order to meet their time constraints and precedence constraints. Finally, an example of application shows the process of scheduling of all real-time tasks.

**Keywords:** real-time, task, communication overtime, compact mode, job rate monotonic (JRM)

### 1 引言(Introduction)

Foundation Fieldbus(FF)是由 Fieldbus Foundation 推出的现场总线协议. 它是一个连接现场设备, 如传感器、执行器和控制器的全数字、多节点、串行、双向的通信系统, 被称为过程和制造自动化领域的计算机局域网<sup>[1]</sup>. 1999 年, 该协议成为现场总线国际标准 IEC61158 八种总线之一<sup>[2]</sup>. 它是目前在国际上比较有发展前途的现场总线之一, 对 FF 协议的研究已经成为工程界和学术界研究的重点和热点.

FF 现场总线系统将控制功能下放到低层现场设备的功能块 FB(Function Block)中, 真正实现了

网络环境下的分布式过程控制. 同时, 对有些通信信息, 如用于不同设备中功能块输入/输出数据缓冲区刷新的远程周期性信息, 提出了很高的实时通信要求, 即在确定的、有限的时间内有效地完成信息的通信, 以保证系统的正常运行.

作为分布式实时控制系统, FF 系统需要相关的硬实时调度算法来解决功能块和远程周期性信息的调度. 在 FF 现场总线系统中, 信息的调度是非抢占式的, 即信息一旦开始在总线上发送, 必须发送完, 才允许其它高优先级或低优先级的信息在总线上发送. 许多学者已经对非抢占式的硬实时调度理论做了大量的工作. Xu<sup>[3]</sup>解决了具有释放时间、截止期、

\* 收稿日期: 2002-01-20  
基金项目: 国家自然科学基金(60084001)资助

次序和互斥约束的多处理器中任务的调度问题,但是没有考虑不同处理器中任务间信息通信对任务调度的影响. T. F. Abdelzaher 和 K. G. Shin<sup>[4]</sup>考虑了不同处理器间任务的信息通信,研究了通信信息和任务的调度,但是假设信息的延迟是固定的且已知的. Tovar, 王和周<sup>[5-7]</sup>首先详细地研究了现场总线系统中的信息调度,做了开创性的工作. 他们分别将单网段的 WorldFIP 和 FF 现场总线系统中信息的调度视为单处理器中任务的调度,并假设所有的周期性信息是独立的,采用基于临界时刻的调度方法. 然而,在由多个相互通信的功能块构成的 FF 现场总线系统中,信息的通信通常要考虑各功能块执行顺序所引起的次序约束. 这意味着,所有的信息不可能同时发出占用总线的请求. 因此,基于临界时刻的信息调度是保守的、不合理的. Zhou 等<sup>[8]</sup>通过调整远程周期性信息的释放时间和截止期来考虑和描述它们的次序约束,但是假设功能块的开始执行时刻是固定的且已知的. 由于在设置功能块的开始时刻没有充分考虑信息的通信情况,这可能会导致有些信息是不可调度的.

本文研究了单网段 FF 现场总线系统中具有时间约束和次序约束的实时任务的调度,即功能块任务和通信任务的调度,并通过一个应用实例来描述它们的调度过程. 本文的主要贡献在于: (1) 在对功能块任务和通信任务建模和调度时,没有将二者完全分开考虑,而是将它们等视为相同的任务; (2) 为了保证每个作业尽可能早地完成,只考虑同一作业中任务的次序约束,提出了基于紧凑模式的任务模型; (3) 考虑通信任务共享一个传输介质而可能引起的通信超时,提出了基于作业速率单调优先级的扩展紧凑模式的任务调度方法,以满足它们的时间约束和次序约束.

## 2 实时任务模型(Real-time tasks model)

在工业过程控制中,通常由多个相互关联或相互独立的控制回路组成,以完成系统的某种要求. 作为新一代的工业控制系统,FF 现场总线系统是将控制功能下放到相互联接的现场设备的功能块中,用户只需通过组态,即确定功能块的类型、位置和它们间的连接,便构成控制系统. 同时,所有的现场设备有一个共同的同步时钟,从而,它们能够循环地按照一个确定的次序调度其功能块的执行和在确定的时刻发布它们的数据. 一个设备中的功能块可以应用

其它设备中功能块发出的远程信息,这样,该功能块也只有在接受这些信息后才能被执行<sup>[9]</sup>. 可见,为了保证控制系统的性能要求,功能块和远程周期性信息的执行具有实时要求.

如图 1,为对一个串联回路的现场设备组态. 它是将 5 个功能块(AI1, AI2, PID1, PID2, AO),下放到 3 个现场设备中,并通过 3 个外部连接(data1, data2, data3)和 3 个内部连接构成一个串联回路. 其中,每个外部连接对应一个远程周期性信息,它需要周期性地在线上被实时传递;内部连接对应本地周期性信息,它在本地被处理,不需要在总线上传输,本文对此不作研究. 另外,本文对于实时性要求不高的非周期性通信信息也不作研究,有兴趣的读者可参见文献<sup>[10]</sup>.

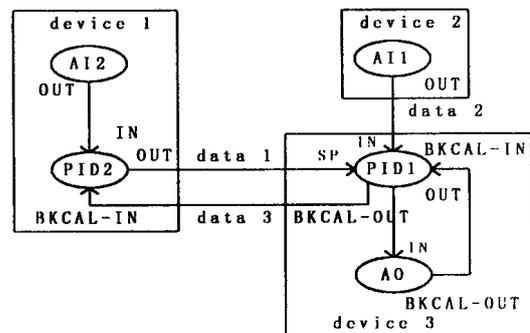


图 1 一个串联回路的现场设备组态

Fig. 1 A cascade loop field devices configuration

本文将现场设备和总线视为处理器,前者处理功能块,后者处理通信信息. 相应地,功能块和远程周期性信息被分别称为功能块任务和通信任务. 可见,一个控制回路是由一些具有次序约束的任务构成的,任务是一个控制回路中被执行的基本单位.

为了便于描述,需介绍下面的五个基本概念:

定义 1 作业[*i*](Job[*i*]), 它对应于控制回路[*i*], 是控制回路[*i*]中具有次序约束的所有任务 task 的集合. Job[*i*]被描述为

$$\text{Job}[i] = \{ \forall (\text{task}_j[i] < \text{task}_k[i]) \\ \text{task}_j[i], \text{task}_k[i] \in \text{control loop}[i] \} \quad (1)$$

其中,  $\text{task}_j[i] < \text{task}_k[i]$  表示 Job[*i*]中的任务  $\text{task}_j[i]$  和  $\text{task}_k[i]$  有次序约束,即  $\text{task}_j[i]$  执行完成后才能执行  $\text{task}_k[i]$ . 对于图 1, AI1 是一个任务,这个控制回路对应的 Job 被描述为

$$\text{Job} = \{ (\text{AI2} < \text{PID2}), (\text{AI2} < \text{data2}), \\ (\text{PID2} < \text{data1}), (\text{data1} < \text{PID1}), \\ (\text{data2} < \text{PID1}), (\text{PID1} < \text{AO}), \\ (\text{PID1} < \text{data3}) \}$$

**定义 2** 通信超时(Communication Overtime), 它是一个通信任务遭受其它作业中的通信任务的直接或间接的阻碍, 使得此任务不能在其释放时刻被执行, 而被延迟执行且超过了其截止期的现象.

**定义 3** 及时可调度通信任务(Timely Schedulable Communication Task), 它是能够在其释放时刻被立即传递的通信任务.

**定义 4** 紧凑模式(Compact Mode), 它是一个紧挨着一个地确定任务释放时间和截止期的算法, 其思想是前序任务执行结束后尽可能早地执行后序任务, 即后序任务的释放时间不能超过其前序任务的执行结束时间; 同时前序任务的截止期不能滞后于其后序任务的释放时间. 这样, 使得 Job[*i*]在控制回路[*i*]的一个控制扫描时间间隔(控制回路[*i*]的周期)内被尽可能早地执行完毕.

**定义 5** 作业 Job[*i*]的理想执行结束时间(Ideal Finish Time of Job[*i*]), 它是不考虑 Job[*i*]中的通信任务遭受的通信阻碍, 即假设通信任务释放后能被立即执行, 按照紧凑模式, Job[*i*]中最后执行完毕任务的结束时间.

**定义 6** 宏周期(Macrocycle), 它是单网段所有远程周期性信息周期的最小公倍数.

众所周知, FF 协议采用集中式通信介质访问控制方式, 即在每个网段有一个链路活动调度器 LAS(Link Active Scheduler)按照一个预定的调度时间表, 控制通信信息在总线上的传递<sup>[1]</sup>. 宏周期就是这个循环间隔, 决定了这个调度时间表的规模且充分反应了功能块和周期性信息的调度情况.

不失一般性, 考虑一个由  $n$  个现场设备构成  $m$  个控制回路的 FF 现场总线网段. 每个控制回路[*i*]对应一个 Job[*i*],  $i = 1, \dots, m$ , 有  $mp_i$  个功能块任务和  $np_i$  个通信任务. 当然, 在同一个作业中的所有任务的周期是相同的, 等于 Job[*i*]的周期, 即控制回路[*i*]的控制扫描时间  $T[i]$ . 假设每个作业[*i*]中任务  $k$  的最大执行时间  $C_k[i]$  是已知的. 只考虑次序约束, 而不考虑通信阻碍. 在这种情况下, 按照定义 4 的紧凑模式, Job[*i*]中任务  $task_{j[i]}$  的首次释放时间  $R_{j[i]}$  为

$$R_{j[i]} = \begin{cases} \max\{R_k[i] + C_k[i]\}, & \forall K: task_k[i] < task_{j[i]} \\ & k \neq j, task_k[i] \in Job[i] \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

这样, 由式(2)确定了作业 Job[*i*]中所有任务的首次释放时间后, 所有任务的首次截止期为

$$D_{j[i]} = \begin{cases} \min\{R_k[i]\}, & \forall K: task_j[i] < task_k[i] \\ & k \neq j, task_k[i] \in Job[i] \\ T[i], & \text{otherwise} \end{cases} \quad (3)$$

由定义 6, 在作业 Job[*i*]中的每个任务  $task_{j[i]}$  在一个宏周期内将被调度  $n_{j[i]} = \text{macrocycle}/T[i]$  次, 其中任务  $task_{j[i]}[l]$  为  $task_{j[i]}$  的第  $l$  次调度. 因此, Job[*i*]的任务模型被描述为

$$task_{j[i]}[l] = (R_{j[i]}[l], C_{j[i]}, D_{j[i]}[l]) \quad (4)$$

其中,

$$\begin{cases} R_{j[i]}[l] = R_{j[i]} + (l-1)T[i], \\ D_{j[i]}[l] = D_{j[i]} + (l-1)T[i], \end{cases} \quad l = 1, \dots, n_{j[i]}.$$

在一个宏周期内被调度任务的总数为,

$$n_{\text{total}} = \sum_{i=1}^m \sum_{j=1}^{mp_i+np_i} n_{j[i]} = \text{macrocycle} \times \left( \sum_{i=1}^m \sum_{j=1}^{mp_i+np_i} \frac{1}{T[i]} \right) \quad (5)$$

### 3 调度算法(Scheduling algorithm)

#### 3.1 优先级分配

通常, 根据任务的特征, 对任务分配不同的优先级, 采用基于优先级的调度是满足任务实时性要求的重要方法<sup>[11]</sup>. 本文, 对作业中通信任务的调度也采用基于优先级的方法. 这里, 作业的截止期等于作业的周期.

**定义 7** 作业 Job[*i*]的松弛时间(Slack Time of Job[*i*]), 它是 Job[*i*]的周期与其理想执行结束时间的偏差, 即

$$T_s[i] = T[i] - E[i], \quad i = 1, \dots, m, \quad (6)$$

其中, 由定义 5,  $E[i]$  是 Job[*i*]的理想执行结束时间, 即 Job[*i*]中最后一个任务的执行结束时刻, 它被描述为

$$E[i] = \max\{(R_{j[i]} + C_{j[i]})\}, \quad \forall j: \exists \bar{k}: \\ task_{j[i]} < task_{\bar{k}[i]}, \quad k \neq j, \\ task_{\bar{k}[i]} \in Job[i] \quad (7)$$

**算法 1** 作业速率单调算法 JRM(Job Rate Monotonic): 对作业进行优先级分配时, 作业的周期越小, 优先级越高; 周期相同时, 作业的松弛时间越少, 优先级越高. 即

$$(T[i] < T[j]) \cup [(T[i] = T[j]) \cap (T_s[i] < T_s[j])] \Rightarrow Job[i] < Job[j] \quad (8)$$

其中,  $Job[i] < Job[j]$  表示  $Job[i]$  的优先级高于  $Job[j]$ . 这意味着, 在高优先级作业中的任务被优先调度, 相应地, 它的通信任务也比低优先级的通信任务被优先分配总线资源. 所以, 这个算法也是通信任务的优先级分配算法.

### 3.2 基于优先级的扩展紧凑模式的任务调度算法

由定义 2, 当一个通信任务被释放时, 由于更多的总线资源已经被分配给其它具有更早释放时间或更高优先级的通信任务, 使得总线正在传递其它的通信任务, 或者在当前时刻没有足够的总线资源来传递此通信任务, 使得此通信任务将不得不等待. 当获得有效的总线资源时, 其截止期可能被错过, 从而发生了通信超时. 这时, 受阻的通信任务和其后序任务的释放时间和截止期将被调整, 于是提出了基于作业速率单调优先级的扩展紧凑模式的任务调度算法. 它是在考虑通信超时的情况下, 使得高优先级的作业能够被尽早地执行, 以便满足作业的实时性要求, 即作业中最后任务的执行结束时间不能超过此作业的周期.

**算法 2** 基于作业速率单调的扩展紧凑模式 (Extended Compact Mode Based on JRM): 按照作业优先级和前面所提的紧凑模式来调度作业中的所有任务, 高优先级作业的任务被优先调度. 当某一通信任务遇到通信超时而不能满足式(3)规定的截止期要求时, 将调整其截止期到它传递结束时刻, 同时其后序任务的释放时间和截止期也要进行相应的调整, 调整的方法也是基于定义 4 的紧凑模式.

考虑通信超时, 次序约束和时间约束, 本文采用基于作业速率单调优先级的扩展紧凑模式的算法对单网段 FF 系统中所有的实时任务进行调度. 此调度算法实现的伪码如下

```
// i is index of Job, from 1 to m
sort Job according to their priorities; // priority of Job is based
// on JRM
for i= 1 to m // Process Job[i]
{ n[i]= macrocycle/T[i]; // Job[i] should be executed n[i]
// times in a macrocycle
CurrentTime= 0; // set init time of macrocycle is zero
for l= 1 to n[i] // the lth schedule for Job[i],
// 1 ≤ l ≤ n[i]
{ for j= 1 to (mpi+ npi) // j is index of a task in Job[i],
// from 1 to (mpi+ npi);
{ // sort task to make taskj[i][l] has the earliest
// release time
sort taskj[i][l] in Job[i] according to their release
time Rj[i][l] from j to (mpi+ npi);
```

```
if taskj[i][l] is a FB task or a timely schedulable
communication task at the current time then
{ Sj[i][l]= Rj[i][l]; // Sj[i][l] is the start
// time of taskj[i][l]
CurrentTime= Sj[i][l];
}
else // deal with the communication task which can
// not be scheduled at the current time
{ for t= CurrentTime to (T[i]' / l)
{ t+ + ;
if the task can be executed at t then exit this
cycle
}
CurrentTime= t;
Sj[i][l]= CurrentTime;
Offset= Sj[i][l]+ Cj[i]- Dj[i][l];
if Offset> 0 // Adjust the deadline of taskj[i][l]
{ Dj[i][l]= Dj[i][l]+ offset;
// Adjust release times and deadlines of
// successors of taskj[i][l]
// for all successors of taskj[i][l]
{ Rk[i][l]= max{Rk[i][l], Dj[i][l]};
Dk[i][l]= max{Dk[i][l], (Rk[i][l]+ Ck[i][l])}
∀ k: taskj[i][l] < taskk[i][l]
}
}
}
FinishTime[i][l]= Sj[i][l]+ Cj[i]; // the finish
// time of taskj[i][l]
if FinishTime[i][l]> T[i]' l
FAILED;
} // for task
CurrentTime= T[i]' (l- 1);
} // for
} // for Job
```

最后, 通过此调度方法确定了所有任务的开始执行时间. 这样, 所有的通信任务的开始时间构成了调度时间表, 被提交给 LAS; 功能块任务的开始执行时间被提交给 FF 的系统管理.

值得一提的是, 按照此调度算法, 每个作业的完成时间不超过其所对应的控制回路的控制扫描时间, 则所有的任务是可调度的, 且本文所提的调度算法是有效的. 否则, 需要将单网段变成多网段, 或者对这个调度算法进一步优化, 这将在下一步的工作中进行研究.

## 4 应用实例 (An example of application)

假设一个单网段的 FF 系统, 如图 2 所示, 由位于 9 个现场设备中的 14 个功能块构成 4 个独立的作业(控制回路), 其中, 有 8 个通信任务需要周期

性地实时通信. 如前所述, 考虑每个作业中每个任务的执行次序和执行时间, 按照式(3) 和(4), 表 1 为

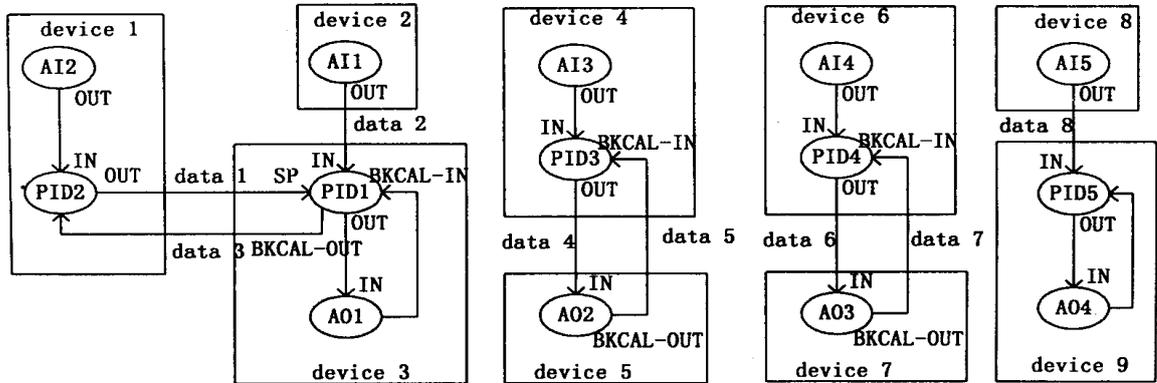


图 2 一个单网段 FF 系统的组态

Fig. 2 A single segment FF system configuration

表 1 任务的时间参数

Tab. 1 Times parameters of tasks

任务	执行时间 $C_{f[i]}(ms)$	首次释放时间 $R_{f[i]}(ms)$	首次截止期 $D_{f[i]}(ms)$	周期 $T[i](ms)$	理想的结束时间 Job(ms)	作业的松弛 时间(ms)
Job[ 1 ]	A I2	30	0	30	210	90
	PID2	65	30	95		
	Data1	20	95	115		
	A I1	35	0	35		
	Data2	20	35	115		
	PID1	65	115	180		
	Data3	20	180	300		
Job[ 2 ]	AO1	30	180	300	130	70
	A I3	30	0	30		
	PID3	50	30	80		
	Data4	15	80	95		
	AO2	20	95	115		
Job[ 3 ]	Data5	15	115	200	155	45
	A I4	35	0	35		
	PID4	60	35	95		
	Data6	20	95	115		
	AO3	20	115	135		
Job[ 4 ]	Data7	20	135	200	100	0
	A I5	20	0	20		
	Data8	15	20	35		
	PID5	43	35	78		
AO4	22	78	100			

这样, Job[ 4 ]-Job[ 3 ]-Job[ 2 ]-Job[ 1 ]就是按照 JRM 算法的作业优先级递减排序. 那么, 应用本文所提的调度算法, 在一个宏周期内所有实时任务的调度过程如图 3 所示.

### 5 结论(Conclusion)

本文研究了单网段 FF 现场总线系统中具有时间约束和次序约束的实时任务的建模与调度. 首先,

将功能块任务和通信任务视为相同的任务, 只考虑任务间的次序约束, 提出紧凑模式的任务模型, 以保证每个作业尽可能早地完成. 其次, 考虑单网段通信任务共享一个传输介质而引起的通信超时, 提出了基于作业速率单调优先级的扩展紧凑模式的实时任务调度算法, 以满足它们的时间约束和次序约束. 最后, 通过一个应用实例来描述它们的调度过程, 为实际工程应用提供指导.

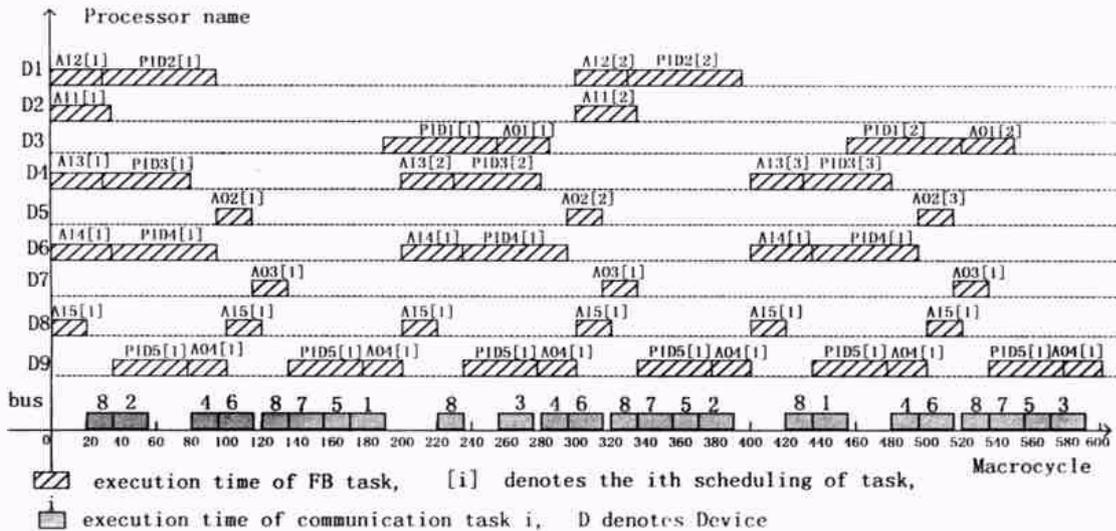


图3 实时任务调度过程

Fig. 3 Scheduling process of real-time tasks

参考文献 (References)

- 1 阳宪惠. 现场总线技术及其应用. 清华大出版社, 1999
- 2 熊菊秀等. 现场总线及其发展展望. 世界仪表与自动化, 2000, 4 (18): 44~ 47
- 3 Jia Xu, Multiprocessor Scheduling of Processes with Release Times, Deadlines, Precedence, and Exclusion Relations, IEEE Trans. on Software Eng. 1993, 19(2): 139~ 154
- 4 Tarek F. Abdelzaher and Kang G. Shin, Combined Task and Message Scheduling in Distributed Real-Time Systems, IEEE Trans. on Parallel and Distributed System, 1999, 10(11): 1179 ~ 1191
- 5 Eduardo Tovar, Supporting Real-Time Communications with Standard Factory-Floor Networks, Portugal: Polytechnic Institute of Porto, the dissertation for the degree of Doctor, July 1999
- 6 王 智. 面向现场总线的分布式实时系统的建模与分析方法. 中国科学院博士学位论文, 2000
- 7 周 悦, 王 智, 于海斌, 王天然. 扩展单调速率算法及在 FF 调度时间表构建中的应用. 信息与控制, 2001, 30(5): 418~ 421

- 8 Yue Zhou, Zhi Wang, Hai B. Yu and Tian R. Wang, A new method to the FF schedule time list, IFAC Conference on NTCC2001, 539~ 543, 19~ 22 November 2001
- 9 Claudio Aun Fayad, Process Control Performance on a Foundation Fieldbus System, ISA EXPO/2000, 21~ 24 August, 2000
- 10 Zhi Wang, Yue Zhou, Youxian Sun, Tian-ran Wang, Response time of urgent aperiodic message in FF, IFAC ON NTCC2001, 527~ 532. 19~ 22 November 2001
- 11 Giorgio C. Buttazzo, Hard real-time computing systems predictable scheduling algorithm and application, KLUWER ACADEMIC PUBLISHERS, 1997

作者简介

王天然(1943- ), 男, 研究员. 研究领域为机器人控制、分布式控制系统、先进制造等.

周 悦(1970- ), 女, 博士生. 研究领域为分布式控制系统、智能控制等.

于海斌(1964- ), 男, 研究员. 研究领域为智能生产调度、分布式控制系统、离散事件动态系统等.