

文章编号: 1002-0411(2002)06-513-05

FF 现场总线系统调度问题的研究

蒲 维 邹益仁

(中科院自动化所综合自动化技术工程中心 北京 100080)

摘 要: FF 现场总线系统(FCS)为典型的分布式实时系统,分布在不同设备(包括总线)的功能块和通讯任务相互作用完成复杂的控制方案,除了实时性要求,还要考虑执行顺序和资源约束,调度问题为一 NP-完全问题.本文分析了 FF 控制系统的特点,提出了类似于作业车间调度问题(JSSP)的调度模型,针对特殊的模型,设计相应的编码和解码规则以及性能指标,用遗传算法在满足上述约束下构建调度表,实现无抖动调度,最后分析了该方法下系统的可调度条件.

关键词: FF 现场总线系统; 分布式实时系统; 调度表; 遗传算法

中图分类号: TP13

文献标识码: B

STUDY OF FF FIELDBUS CONTROL SYSTEM SCHEDULING PROBLEM

PU Wei ZOU Yi ren

(ERC of Integrated Automatic Technology, Institute of Automation, Chinese Academy of Sciences Beijing 100080)

Abstract: FF fieldbus control system (FCS) is a kind of typical distributing real time system, the interaction of function blocks and communication tasks which distribute on different device and bus implements complicated control scheme. In addition to constraint of time, constraint of sequence and resource is considered. Scheduling problem is NP-complete problem. According to characteristics of FF system, this paper holds scheduling model is similar to job shop scheduling problem. After designing rules of code/decode and fitness function, scheduling table which is satisfied with all constraints is constructed with genetic algorithm (GA), it can schedule FF system without jitter. Scheduling condition is discussed at last.

Keywords: FF fieldbus control system, distributing real time system, scheduling table, genetic algorithm (GA)

1 引言(Introduction)

FF 现场总线系统是应用在生产现场、在微机化测量控制设备之间实现双向串行多节点数字通信的系统,也被称为开放式、数字化、多点通信的低层控制网络.它在制造业、流程工业、交通、楼宇等方面的自动化系统中具有广泛的应用前景^[1].

FF 现场总线系统为典型的分布式实时系统,分布在现场设备上不同功能块的执行和功能块之间的通讯除了满足实时性的要求外,还得满足执行顺序和资源(总线、现场设备)的约束^[2].实际上,FF 系统的实时调度包括功能块和总线的调度(链路活动调度).

2 FF 系统模型(FF scheduling model)

以一 PID 控制回路组态为例:

功能块的调度: ○ 表示位于现场设备中完成相应的控制功能的功能块.

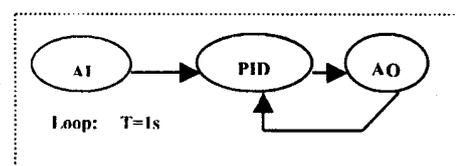


图 1 PID 控制回路

Fig. 1 PID control loop

现场设备按组态预先在系统管理(SM)中的时间调度信息(包括功能块、宏周期、开始时间)循环调度本设备中功能块的执行^[4,5].

总线的调度: → 表示功能块之间的周期性通讯关系,为总线处理的周期性通信任务(如功能块处于同一设备,总线处理通讯任务的时间为 0),该通信

任务是硬实时的(Hard-Time), 必须满足截止期(控制周期)的要求. FF 总线在数据链路层(DLL)上采用集中式令牌介质访问控制方式. 每个网段有且只有一个链路活动调度器 LAS(Link Active Scheduler), 对周期性通信任务(支持 P/S 方式)和非周期性通信任务(支持 C/S 和 R/D 方式)采用不同的调度策略. 对于周期性任务, 按照预先离线形成的时间调度表优先调度周期性任务对总线访问; 对于非周期性任务, 则总线空闲时, 通过发送令牌(Token)调度非周期的通信任务, 如 C/S 和 R/D 通信任务、探测节点(PR)、时间发布(TD)、空闲(Idle)帧.^[3]

FF 总线系统包括功能块和总线的相互作用, 实际的控制方案往往形成复杂的调度关系, 需满足相应约束:

- (1) 时间约束, 每个通信任务和功能块属于周期性任务, 必须在相应的截止期(控制周期)前完成.
- (2) 顺序约束, 功能块之间的关系, 通信任务之间、通信任务同功能块之间存在执行顺序约束关系.
- (3) 资源约束, 通信任务不能同时访问总线, 位于同一现场设备上的功能块不能同时执行.

故不能将其简单的假设为以总线为单处理器的实时模型, 而是多处理器分布式实时系统. 在 FF 系统中, 周期性任务的调度基于静态调度表(Scheduling Table), LAS 根据调度表来管理周期信息通信, 现场设备通过 SM 中的调度表管理功能块的执行. 系统有一时间发布器, 它向所有现场设备周期性的发布应用时钟同步信号, 保证数据链路调度时间和应用时钟的一致, 时间发布器的时间粒度是调度表的最小调度单位, 称为微周期(microcycle), FF 系统的微周期为 1/32 毫秒^[2]. 同一设备所有的周期性信息重复出现的最小时间间隔, 称为宏周期(macrocycle)^[2], 根据 LCM(the Last Common Multiple)方法, 宏周期是所有周期任务周期的最小公倍数, 不同设备的宏周期不一致. 在调度表中包括所有周期任务在宏周期内全部实例的调度情况, 每个周期性任务应包括 $m \text{arcocycle}/T_i$ 个实例, 为了防止任务执行的抖动(Jitter), 实例之间的时间间隔严格为其周期 T_i , 由于无抖动, 实际的调度表只需包括任务调度周期和第一个实例开始时间. 宏周期大小与实际的调度表的规模没有关系, 实际的调度表的规模只取决于控制方案的规模(链路上功能块和通信任务的数量).

3 统一调度模型: (Unified scheduling mod-

el)

FF 系统中存在总线和现场设备的调度, 为了统一处理, 可假设总线为执行通讯功能的现场设备, 周期性通信任务为其执行的功能块. 对于这类调度问题, 可类比于作业车间调度问题(JSSP), JSSP 见文献[8], 即有 m 台不同的现场设备(包括总线)和 n 个不同的的周期性控制回路, 每个控制回路包括一个由若干功能块(包括通信任务)组成的集合, 由于执行顺序由控制回路决定的, 执行功能块的现场设备、时间和周期是确定的, 该模型可解决顺序和资源的约束, 再考虑到到达时间和截止期, 还可满足相应的约束. 该模型与 JSSP 对应的关系为: 机器(现场设备), 工件(控制回路)、工序(控制回路包括的功能块), 不同之处在于:

- 1) 某台机器可处理工件的多道工序, 工件工序的机器表同一行的机器项可能相同, 因此遗传算法的编码方式不同;
- 2) 某工件可能在一些机器上加工(而不是全部机器), 工件工序的时间表是一稀疏阵, 应设计相应的编码方式降低计算的规模, 加快求解的过程.
- 3) 某工件在宏周期内加工若干次, 造成解码方式不同.
- 4) 调度优化的目标不同, JSSP 的优化目标一般是最小化加工时间, 而该模型的优化目标是所有任务满足截止期的要求, 应重新设计适值函数.

4 调度算法(Scheduling algorithm)

在资源、时间和顺序约束下, 且任务非抢占的, 多处理器的调度是完全 NP 问题, 基于启发式的实时调度算法, 如速率单调(RM)、最早截止期优先(EDF)等^[9,10,11], 对于这类分布式系统不可行的, 而在线的优化调度算法由于时间复杂度不太可能实现. 在 FF 系统中, 对于确定的控制方案, 其功能块执行任务和通信任务的特征(执行时间、周期或截止期、执行顺序、访问资源)是确定(可预测), 并且所有任务的执行是以唯一的时钟为基准, 因此离线构建调度是可行的, 同时根据构建的调度判断分布式实时系统的可调度性也是可行的.

FF 系统可离线构建调度表预先分配资源以满足任务截止期的要求, 首先必须知道与调度相关的信息. 周期任务由三个参数描述(C_i, T_i, D_i), C_i, T_i, D_i 分别为周期任务 i 的执行时间、周期和截止期, 由于 $D_i = T_i$, 根据(C_i, T_i)可确定任务的执行. T_i 由控

制方案确定, C_i 对于不同的任务不同, 对于功能块, C_i 由功能块的执行时间(Execution Time)参数确定; 对于通讯任务, C_i 包括两个数据链路层协议数据单元的发送时间、发送两个帧的时间间隔 t_r 之和, 即

$$C_i = \frac{\text{length}(CD + DATA)}{Bps} + 2 \times t_r \quad (1)$$

式(1)中, $\text{length}()$ 表示信息帧的长度, CD 表示 CD 令牌帧, $DATA$ 表示周期信息的数据帧, Bps 表示 FF 的传输速率, 对于 H1 标准, $Bps = 31.25 \text{ kbits/s}^{[3]}$.

接着对给定周期性任务集构造调度表, 分配任务集的子任务到分布式系统的相应的设备上. 在运行前, 将调度表下载到现场设备和 LAS, 在运行时, 设备按各自的调度表以相应的周期和开始时间执行任务, 任务集则按调度表重复执行. 由于 FF 系统调度模型与作业车间模型类似, 对于这类离线静态的组合问题, 可用启发式方法(如分支界定、SA)搜索可行性调度. 本文采用遗传算法(GA)进行求解, 其优化的目标是得到满足时间、顺序和资源约束的静态调度表.

遗传算法是一种基于生物自然选择和遗传机理的随机搜索方法, 本文采用它得到优化的调度结果. 一般地用 GA 求解问题时应考虑如下五个方面: 对于问题的解进行遗传编码; 构造初始染色体; 选择一个适应性函数; 确定遗传算子(选择、交叉、变异)规则; 选择遗传算法的参数(群体中染色体数目、交叉概率、变异概率、最大迭代次数). 下面给出详细的设计^[8].

1) 编码设计

由于本调度问题与作业车间调度问题存在 1)、2) 的不同之处, 本文提出基于工件、工序和机器三元联合表达的方式, 把调度所对应的染色体编码为三者组合的序列, 每个基因可用:

$$\text{Job}^* 10^2 + \text{Sequece}^* 10^1 + \text{Machine}^* 10^0$$

表示, 可保证产生的所有自然数序列都可以定义为可行性调度, 同时不排除任何可行性调度. 对于 n 个

工件, 一个染色体包括所有工序之和即 $\sum_{i=1}^n \text{Sequece}_i$ 个基因, 大大降低染色体的长度和计算的规模.

2) 产生初始种群

从一可行性解集的随机组合产生初始种群, 以可行性解为基础, 防止了非法染色体的产生.

3) 解码设计:

具有同一工件特征的所有基因在序列中的位置决定该工件在调度表内的加工顺序. 但染色体经过交叉、变异操作后, 基因在染色体的位置对于同一工件的工序是乱序的, 需对基因进行整理, 即将同一工件的基因重新排序, 并放在这些基因在染色体相应的位置上. 按整理后的染色体解码成可行性调度表, 以基因在染色体上的位置作为调度顺序. 在分配基因对应的任务前, 判断当前工件前一个工序结束的时间, 以此时间作为当前任务的开始时间, 根据任务的执行时间和周期, 生成任务所有实例执行时间片

$$\cup [kT_i + t_{\text{start}}, kT_i + t_{\text{start}} + C_i]$$

$$k \in \{0, 1, \dots, \text{macrocycle}/T_i - 1\} \quad (2)$$

式(2)中, T_i 为周期, t_{start} 为周期性任务第一个实例的开始时间, C_i 为任务的执行时间, macrocycle 为宏周期的长度. 判断每个时间片和当前该机器已分配的时间片是否存在冲突, 若有, 则已冲突的时间片的结束时间为基础, 修正当前任务的开始时间, 重复上述过程, 直到没有任何冲突或可分配的时间片超过其截止期时结束, 如任务不可分配, 则记录下来.

4) 选择方法和适值函数的定义

优化目标是满足截止期的要求, 期望没有超过截止期的任务, 构造适值函数

$$f_i = e^{-\text{over.deodline.num}_i} \quad (3)$$

式(3)中, $\text{over.deodline.num}_i$ 表示对应第 i 个染色体超过截止期任务的数量. 采用转轮法进行选择, 根据与适值成正比的概率形成新的种群.

5) 交叉和变异

采用循环交叉(CX)、插入变异方法. 交叉和变异中必须保证染色体对应调度的可行性.

6) 停止准则

循环执行进化过程, 以出现 $\text{over.deodline.num}_i = 0$ 染色体或预先设定的最大进化代数为停止条件.

5 可调度条件分析 (Scheduling condition analysis)

在分布式系统中, 对于周期性任务, 当且仅当在宏周期(周期的 LCM (the Last Common Multiple)) 内存在可行性调度时, 任务才是可调度的^[6,7]. 对于单台现场设备, 若考虑由于无抖动非抢占的任务调度方法在顺序和资源约束下造成的阻塞, 通过类似于单处理器下 RM 方法的推导, 仍满足 RM 算法利用率上限, 即设备最大利用率:

$$U_{mb} = \sum_{i=1}^n \frac{C_i}{T_i} + \max_{i=1 \dots n} \left(\frac{B_i}{T_i} \right) < n \times (2^{1/n} - 1) \quad (3)$$

式(3)中, n 为该设备上执行任务的数量, C_i , B_i , T_i 分别为一设备上第 i 个任务的执行时间、阻塞时间和周期^[11]. 实际上, 在静态调度表形成之前是不可能预见到阻塞的情况现场设备, 可先对每个现场设备上执行的任務集按非阻塞情况($B_i = 0$)下可调度条件判定, 若

$$\sum_{i=1}^n \frac{C_i}{T_i} > n \times (2^{1/n} - 1) \quad (4)$$

式(4)中定义见式(3)^[10], 则可判定任务集在该设备上是不可调度的(充分条件), 若满足可调度条件, 则用本文方法构造调度表, 再根据阻塞的情况进行可行性测试, 即可判定系统的可调度性. 若不可调度, 根据不可调度的任务(功能块)情况修改控制方案, 如重新分配现场设备的负荷或减少控制回路数量.

6 调度结果(Scheduling Result)

考虑链路上存在 6 台 Smart 现场设备(包括总线), 其控制组态如图 2.

表 1 设备列表

Tab. 1 List of Devices

序号	1	2	3	4	5	6
设备	LD292	FI302	IF302	FY302	TT302	FieldBus
类别	压力	总线/电流	电流/总线	调节阀	温度	总线

图 2 中包括三个控制回路, 六个设备和相应功能块. 相应的控制周期和执行顺序如组态所示. 按照前述的方法, 先在非阻塞情况下对各个设备进行可调度条件的测试, 接着用 GA 求解, 其中, 种群的数量为 50, 采用 CX 交叉($P_c = 0.6$)和插入变异 $P_m = 0.1$ 方法, 经过 20 代进化, 得到满足要求的调度结果如表 2、表 3 和图 4、图 5 所示.

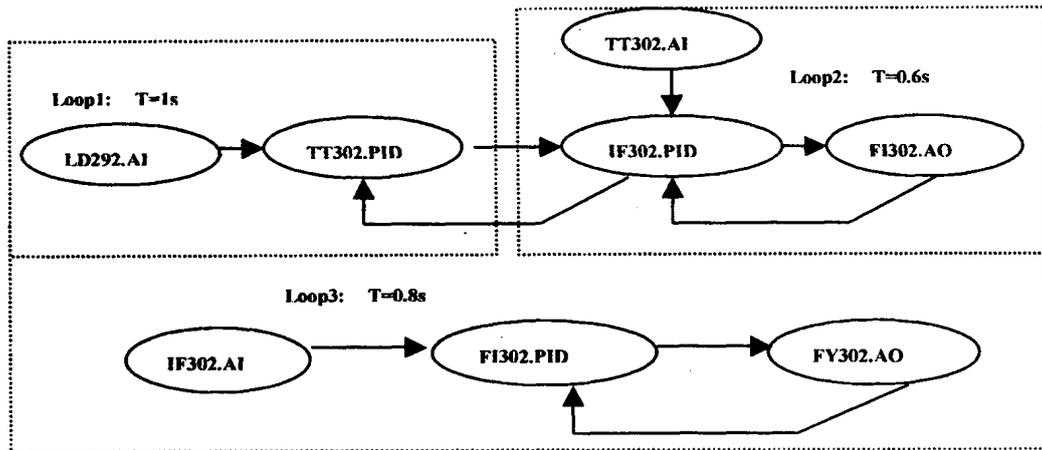


图 2 串级和 PID 控制回路

Fig. 2 Cascade and PID control loop

表 2 功能块调度表(下载到对应设备的 SM)

Tab. 2 Function Block Scheduling Table

功能块	开始时间(1/32ms)	执行时间(1/32ms)	调度周期(1/32ms)
TT.AI(215)	0	1280	19200
LD.AI(111)	0	2560	32000
IF.AI(312)	0	2560	25600
FI.PID(332)	5050	2400	25600
FY.AO(354)	11450	1245	25600
TT.PID(135)	3805	2400	32000
FI.AO(262)	8960	1600	19200
IF.PID(233)	2525	1920	19200

表 3 通讯任务调度表(下载到 LAS)

Tab. 3 Communication Task Scheduling Table

通讯任务	开始时间(1/32ms)	执行时间(1/32ms)	调度周期(1/32ms)
TT. AI→ IF. PID(226)	2525	1245	19200
IF. PID→ FI. AO(246)	5050	1245	19200
IF. PID→ TT. PID(256)	6295	1245	19200
TT. PID→ IF. PID(146)	8925	1245	32000
FY. AO→ FI. PID(366)	16605	1245	25600
FI. AO→ IF. PID(276)	11450	1245	19200
FI. PID→ FY. AO(346)	10205	1245	25600
IF. AI→ FI. PID(326)	3850	1245	25600
LD. AI→ TT. PID(126)	2560	1245	32000

图 3, 4 说明: 纵轴分别表示控制回路和现场设备, 横轴表示时间, 以毫秒为单位, 方框表示执行功能块对应的时间片. 其中 111 表示控制回路 1, 执行顺序 1, 在设备 1(LD292)上的 AI 功能块.

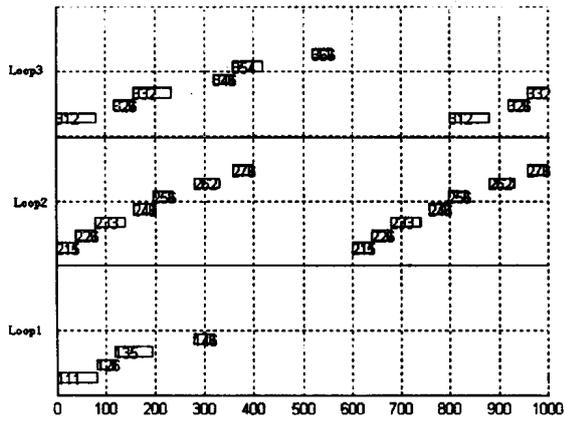


图 3 基于控制回路的甘特图

Fig. 3 Gantt Graph of Control Loop

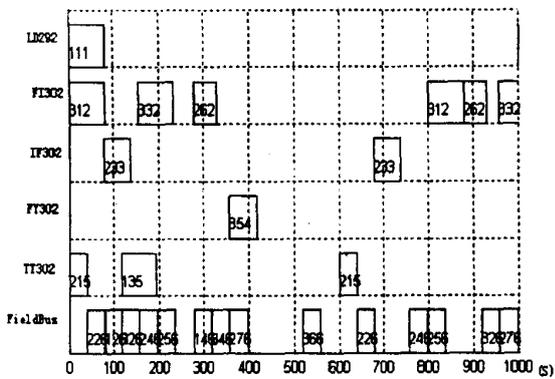


图 4 基于设备的甘特图

Fig. 4 Gantt Graph of Device

7 结论(Conclusion)

在分析 FF 现场总线控制系统的调度过程的基

础上, 提出类比于作业车间的统一实时调度模型, 定义该调度模型的特点, 并根据模型, 设计适合问题的编码、解码方法和适度函数, 用 GA 方法得到可行性调度表, 保证系统的实时性和调度的无抖动, 并给出可调度的条件. 最后, 通过一具体应用实例来说明该调度模型和方法在实际工程中的应用, 并给出相应的调度结果. 还需要对该模型的可调度条件和调度优化作进一步分析和研究.

参 考 文 献 (References)

- 1 阳宪惠. 现场总线技术及其应用. 清华大学出版社, 1999
- 2 FOUNDATIONTM Specification System Architecture. Revision 1. 4, Fieldbus Foundation, 1999
- 3 FOUNDATIONTM Specification Data Link Protocol Specification. Revision 1. 4, Fieldbus Foundation, 1999
- 4 FOUNDATIONTM Specification SM. Revision 1. 4, Fieldbus Foundation, 1999
- 5 FOUNDATIONTM Specification FBAP. Revision 1. 4, Fieldbus Foundation, 1999
- 6 E L Lawler, C U Martel. Scheduling Occurring Tasks on Multiple Processors. Information Processing Letter, 1991, 12(1)
- 7 A K Mok. Fundamental Design Problems of Distributed System for Hard Real Time Environments, PhD Thesis, Lab for CS (MIT), TR- 297, 1983
- 8 [日]玄光男, 程润伟. 遗传算法与工程设计[M]. 北京: 科学出版社, 2000
- 9 Ram aritham G Fohler, J M Adan. Issues in the Static Allocation and Scheduling of Complex Periodic Tasks, 10th IEEE Workshop on Real-Time Operating Systems and Software, May 1993
- 10 C L Liu, J W Layland (1973). Scheduling Algorithms for Multiprogramming in a Hard-Real-Time Environment. Journal of the ACM, 20(1): 46~ 61

(下转第 523 页)